Error and Attack Tolerance of Public Transportation Networks: A Temporal Networks Approach

Arash Badie Modiri

School of Science

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 16.4.2018

Thesis supervisor and advisor:

Prof. Mikko Kivelä

Aalto University
School of Science

AALTO UNIVERSITY
SCHOOL OF SCIENCE

ABSTRACT OF THE
MASTER'S THESIS

Author: Arash Badie Modiri

Title: Error and Attack Tolerance of Public Transportation Networks: A Temporal Networks Approach

Date: 16.4.2018 Language: English Number of pages: 5+54

Department of Computer Science

Professorship: Complex Systems

Supervisor and advisor: Prof. Mikko Kivelä

The behaviour of complex networks under attack provides insight into their internal structure. Furthermore, advances in methods for analysing temporal networks have enabled us to perform more detailed modelling of a certain subset of dynamic complex systems specially since frequency of events and temporal correlations play a role in dynamics of the system. In this report, the temporal network approach for study of robustness is applied to public transportation networks. The focus is on providing a set of tools to model different scenarios of attack and random failure, and processing the results with or without taking into account the origin-destination demand matrix frequently used in transportation network studies.
The results of the robustness analysis on temporal representation of public transport networks illustrate the distribution of accessibility and travel time after an attack or error and how it changes when more routes are removed. Furthermore we see that two methods of attack, one based on temporal betweenness centrality and one based on nominal capacity of routes, have a higher effect on increasing delays while attack methods based on centrality of routes in a static aggregated network do not perform any better than randomly removing routes.

Keywords: Complex Networks, Temporal Networks, Connection Scan Algorithm, Robustness, Public Transportation, Accessibility

Preface

First of all, I would like to thank my supervisor and advisor, professor Mikko Kivalä, for his support and patience.

I would like to thank all my colleges in our research group for their company and support: Mikko, Jari, Kimmo, Tuomas, Sara, Talayeh, Ana, Onerva, Ilkka, Richard, Darko, Gerardo, Rainer, Christofer, Sallamari, Pietro, Daniel, Kunal, and Asim. Surely it has been a true privilege working and socialising with you.

Thanks to Rainer (Kujala) for reading through an earlier version of this document and suggesting many improvements and weaknesses of the original version.

I would also like to thank Professor Gholamreza Jafari and Dr. Amirhossein Shirazi for introducing me to the fascinating world that is complex systems, and professor Jari Saramäki for all his support during my master's studies.

Last but not least, I wish to thank Hasti for always helping out and being there with me through all this, and for lighting a candle when this world began to look dark and gloomy.

Otaniemi, 16.4.2018

Arash Badie Modiri

Contents

1 Introduction

The scientific method, or as Newton would have it "the rules of reasoning in (natural) philosophy", has been leading the natural philosophers on toward a fine line between generalisation of the models on one hand and a more reductionist approach of specialising models for a narrower and narrower set of phenomena on the other. While the body of science at any time mostly focuses on specialising the models, there is always the tendency to take a step back and try to explain a more universal set of phenomena with a simpler model. Most of the attempts have not stood the test of time—the intricate vortex model of gravitation of Descartes and Huygens as an example turned out, like many other similar attempts, to produce few measurable implications and couldn't provide an explanation for many empirical observations—but those that did stand provided a new set of tools for understanding nature.

Complex systems gained popularity among statistical physicists with the idea that a great variety of phenomena, previously largely inaccessible to scientists, can be modelled as a large number of simple entities with simple interactions between entities. Many of the macroscopic behaviours of the system can be calculated and deduced as emergent properties of the model without trying to model every microscopic detail of each entity and every interaction. This was further developed as models that were devised to describe one class of "complex" phenomenon were generalised to other classes of phenomena and scientists could use the same understanding about the higher level behaviour of the system by abstracting away micro-scale internal machinery of each system. As an example, Ising model which started as a model of describing ferromagnetism in statistical mechanics is later on used to study tax evasion [1], financial markets [2] and social interactions [3].

Complex networks, one of the most popular approaches in the study of complex systems, model the phenomena as networks of entities. Developing on tools borrowed from graph theory, it provides insight on some of the emerging properties of the systems based on the structure of the network. In its current form, it can be traced back to studies of the structure of empirical networks and identifying certain common shared features described by terms such as small-world [4] and scale-free [5]. This type of non-trivial structure can be attributed to many of the emerging properties of the family of real-world phenomena studied using networks.

One of the methods employed in studying the structure of networks is the method of robustness, based on percolation theory from statistical physics, where behaviour and properties of the network are studied after some of its parts (interactions or entities) are removed. This, apart from insights on the roles of structural properties and inhomogeneities of the network in its connectivity, gives us an insight into the reasons behind the stability of many of these real-world systems.

In the same way that non-trivial structural properties of "static" networks contribute to some of their behaviour, it was shown that for some phenomena there are non-trivial temporal structures that affect the behaviour of the system and not taking them into account might result in an incomplete modelling of the phenomena [6, 7]. It has also been shown that temporal correlations and inhomogeneities of the phenomena affect the speed of spreading processes on the network [8]. By retaining temporal information, timestamps and durations, of interactions, *Temporal network* model tries to provide a more accurate model of the phenomena by embodying these temporal inhomogeneities and structures.

Modelling of public transportation systems as static networks in the last decade has lead to insights into their internal structure and emerging properties [9, 10, 11, 12, 13, 14] but temporal modelling of public transportation systems is still comparatively rare. Throughout the rest of this Thesis, we provide methods and algorithms to measure quantities that are important in operation of public transportation systems from a *temporal network representation.* These quantities include accessibility and travel time throughout the public transportation system. We then proceed to perform robustness analysis on the temporal representation of multiple public transportation networks and measure the effects of these quantities. By comparing the effectiveness of different ways of removing parts of the network we can learn which method is more successful at finding routes that are more important in overall connectivity of the network.

1.1 Outline of this Thesis

Chapter 2 Complex Networks is dedicated to a general background on complex networks. We introduce static and temporal networks, various statistics used to describe them, and the concepts of percolation and robustness.

Chapter 3 Material and Methods concerns the data, pre-processing and our implementation of robustness and different attack and error methods for temporal networks and specifically public transportation networks. We also discuss the main algorithm behind the travel time measurements and some of the practicalities of using the software developed to perform the attacks and measure their impact. We also construct a static version of network and measure the effectiveness of using centrality measures of that static network as an attack method on the temporal representation.

In Chapter 4 Results we present the results of the experiments, the impact of different attack methods on the travel times and accessibility. We see that removal methods calculated from the static aggregate representation of the network fail to perform any better than randomly removing routes while methods calculated from the temporal representation of the network perform considerably better at reducing accessibility and increasing travel times.

Chapter 5 Discussion is dedicated to a higher level analysis of the results and their implications, while also deliberating on the limitations of the study and some of the possible methods of going beyond what is presented here for a deeper understanding of public transportation networks.

2 Complex Networks

In order to study the connectivity and structure of public transportation networks, it is imperative to model the system in a way that puts more weight on the connectivity of the system. Network theory provides a good foundation for such a model while still giving the scientist a range of modelling apparatus with varying levels of simplicity. Network theory also provides a set of tools, in form of percolation theory and robustness analysis, to study resilience of the phenomenon or "importance" of elements in the structural integrity and connectivity of the system.

Networks have been used to model many different types of phenomena, especially phenomena having to do with many heavily inter-connected entities, in different fields of science [15]. A *network*, is a set of entities (*nodes* or *vertices*) with connections (*links* or *edges*) that can exist between pairs of nodes [16].

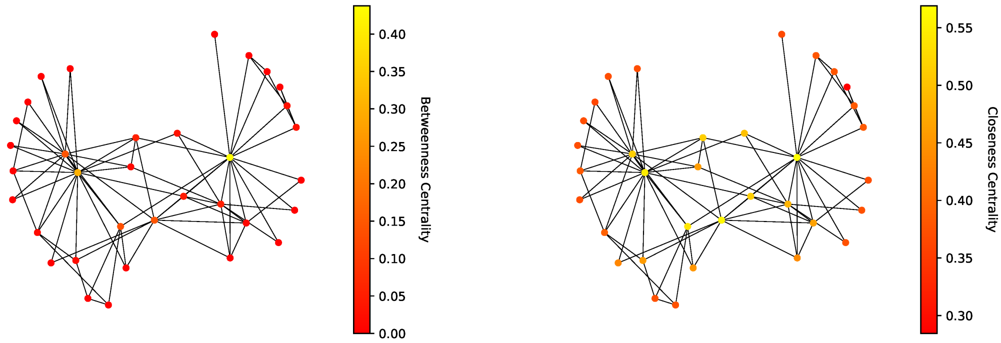
Details of the representation of phenomenon depend on the nature of the system that is being modelled, questions the researchers want to be answered and the degree of accuracy they need to reproduce the real-world measurements with their models: nodes or links might carry additional information, e.g. labels, types, layers or weights, links might have directionality and the existence of nodes and links might be a function of time. This chapter is dedicated to a more detailed exploration of two different types of networks and their properties.

2.1 Static, Undirected Networks

Static networks, commonly referred to as *graphs* in mathematics literature, is defined as a tuple (V, E), where V is a set of nodes (or vertices) and E a set of links (or edges) and each link consists of an (ordered or unordered) pair of nodes [17]. Networks can also be represented by an adjacency matrix A where A_{ij} is 1 (or equal to link weight in case of a weighted network) if there is a link between nodes i and j and otherwise 0 [18]. Also, two nodes that share a link are called *adjacent* nodes.

A static, undirected network is a static network where each link is an unordered pair of two nodes or in terms of adjacency matrix either a two-sided relationship exists between two nodes or not, i.e $A_{ij} = A_{ji}$, which results in a symmetric adjacency matrix.

A static, undirected network is particularly useful when modelling the evolution of connectivity as a function of time is not an objective of the study and when connections between the entities do not have an intrinsic directionality. An example of this is the network of co-authorship in academia; each scientist can be modelled as a node and every two scientists that have published a paper together are connected with a link [19]. If we are not interested in the evolution of the relations between scientists over time, we can model this as a single static, undirected network since co-authorship does not have a specific direction. Figure 1 shows an example of a static, undirected network observed from a real-world social network.



(a) (b)

Figure 1: A well-known network based on real-world data often called "Zachary's karate club" recorded by long-term observation of social relations between members of a sport club [20]. Members of the karate club are represented as nodes and social relationship between them as links. Two members are connected if they "consistently interacted in contexts outside those of karate classes, workouts, and club meetings." In Panel a nodes are coloured according to their betweenness centrality while in Panel b according to their closeness centrality.

2.1.1 Degree distribution and scale-free networks

The *degree* of a node is the number of nodes it is adjacent to. *Degree distribution* $p(k)$ is the fraction of nodes with degree k in the network, which is equal to the probability of selecting a node uniformly at random and coming up with one that has degree k [18]. Degree distribution is usually visualised by forming a histogram of degrees of nodes or some form of density estimation.

In some random graphs, generated through the model suggested by Paul Erdős and Alfréd Rényi [21], degree distribution is binomial. It approaches a Poisson distribution when the number of nodes is increased with the average degree kept constant. [22]. This means that the tail of the degree distribution (as we increase the degree toward larger numbers) declines exponentially. On the other hand, many networks modelling real-world phenomena, such as the world airline network [23], the backbone of the Internet [24] and World Wide Web [25], are shown to have heavy-tailed degree distributions, distributions whose tails are not exponentially bounded. This property of having a significant fraction of nodes with very high degrees has implications in connectivity, vulnerability and resilience of the network against a random failure which will be discussed in more details in section 2.1.6. Networks whose degree distributions (at least for larger values of degree) follow a power-law, defined as $p(k) \sim k^{-\gamma}$, are called scale-free networks [5].

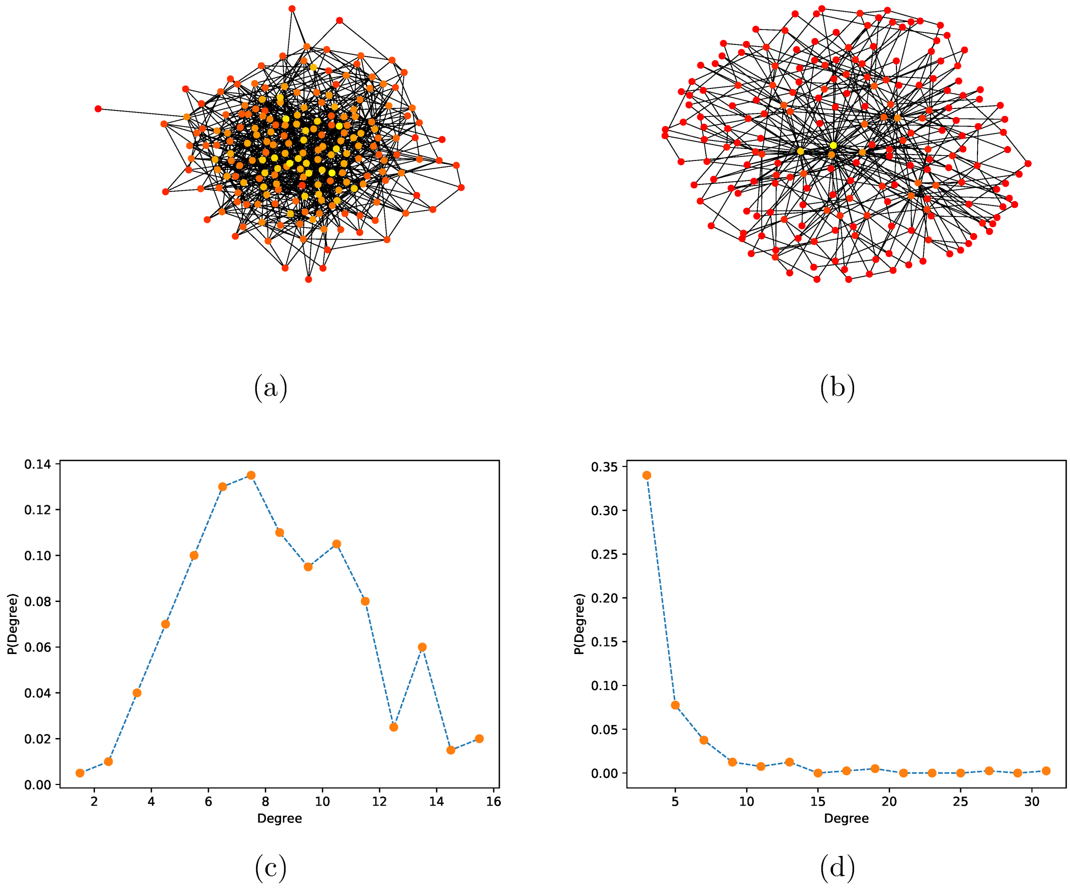
The three networks listed above are reported to be scale-free based on the degree distributions obtained by observation of real-world phenomena in their respective papers. Some have reported [26] results of analysis based on more rigorous statistical methods that cast doubt on the universality of scale-free properties of many networks based on real-world phenomena without providing a more accurate or simpler model. These reports have been met with conceptual and technical criticism. As an example, it has been pointed out that this method fails to classify artificial networks as “strongly scale-free” for which there exists a formal exact proof of scale-free property [27].

2.1.2 Paths and distance

In order to study connectivity and reachability on networks, one needs to rigorously define the concepts related to them. In network theory, defining *path* would be a good starting point since all the concepts related to connectivity can be derived from it.

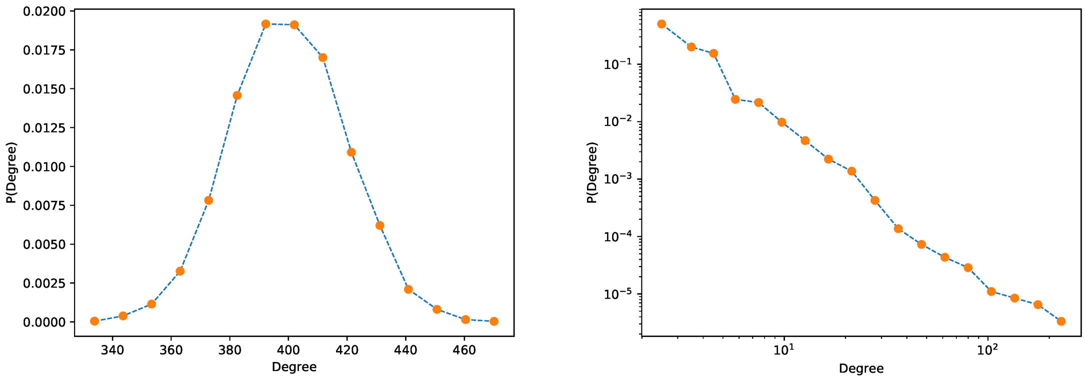
A *path* between two nodes in a static, undirected network is a sequence of nodes beginning with one of the nodes in question and ending with the other, where each pair of consecutive nodes on the sequence are connected by a link. If there exists at least one such path between two nodes, the two nodes are considered *connected* and connectedness of two nodes in a static, undirected network is a symmetric and transitive relationship [18]. Depending on the phenomena at hand, *path length* can simply be the number of links in the path or in case of weighted networks, an aggregation of a function of link weights (e.g. sum of link weights or sum of reciprocals of link weights) for all the links in the path [15].

Shortest path, or *geodesic path*, is the path between two nodes such that no shorter path exists [18]. Shortest path length between two nodes, or the *distance*, is the



,

Figure 2: Two randomly generated networks and their degree distributions. Panels a and c are presentation and degree distribution of a random network generated with an independent probability of existence for every possible edge; one of the two methods named after Paul Erdős and Alfréd Rényi. Panels b and d are presentation and degree distribution of a random network generated through model proposed by Albert-László Barabási and Réka Albert where the probability of receiving new links is proportional to current degree. This method, preferential attachment, creates inhomogeneity in degrees among nodes where there is a much higher chance of nodes with a very high degree compared to an Erdős-Rényi network with the same number of nodes and links. Both networks comprise of 200 nodes. Edges in Panel a exist with probability 0.04 while in Panel b each node adds two new links to the network.



(a) (b)

Figure 3: Degree distributions for two types of randomly generated networks. Panel a is degree distribution of a random network generated according to Erdős-Rényi model with each edge having a 0.04 probability of existence. Panel b shows degree distribution of a random network generated through model proposed by Albert-László Barabási and Réka Albert with each new node adding two new edges to the network. Both networks have about 20000 edges and 10000 nodes. Note that Panel b is plotted on a log-log scale and is more or less linear, implying a power-law distribution of degrees, while Panel a shows a Poisson-shaped distribution.

length of that path. In static, undirected networks, the shortest path lengths between nodes are symmetric and can be calculated using Dijkstra's algorithm [28]. In the special case of unweighted networks, Dijkstra's algorithm simplifies to breadth-first search algorithm. This algorithm will be generalized to more complicated network models needed for analysing public transportation networks in Section 3.2.2.

To characterise connectivity of a network, one can calculate the full distribution of path lengths between all pairs of nodes. However, sometimes it is useful to resort to simpler summary statistic, i.e. single numbers describing some aspects of the connectivity. The *diameter* of the network is of such statistic defined as the longest shortest path length in the network, which is undefined for networks that have pairs of nodes not connected to each other. *Average shortest path length* of a network is defined as the average of shortest path lengths between every pair of nodes in the network.

A *connected components* is maximal sets of nodes in which every node is connected to all the other[18]. Based on this definition and the definition of connectedness, it is clear that connected components are disjoint. *Largest connected component* is found based on the cardinality of each set, i.e. by finding the set with the maximum number of nodes belonging to it. In the study of diameter and average shortest path length, most of the time only the largest connected component of the network is considered.

2.1.3 Small-world networks

Connectivity properties of regular lattices have been analytically studied in the context of statistical physics [29]. However, both real-world and completely random networks have much shorter path lengths compared to a regular lattice with comparable average degree and node count. Watts and Storagtz showed that adding a very small number of random links is enough to make the path lengths much shorter even if you start from very structured network with large path lengths such as lattices [4].

Many networks, artificial or observed from real-world phenomena, have a very short average shortest path length compared to regular lattices of the same size and average degree despite most nodes not being directly connected to each other. If the average shortest path length of a network scales as $\sim \log(n)$ or slower, where n is number of nodes in the network, the network is small-world [16]. Average shortest path in random networks that follow a power-law degree distribution with exponent $2 < \gamma < 3$ are shown scale as $\sim \log\log(n)$ [30].

It is important to distinguish between this statement and observation of growth in a single real-world network. When observing growth in a single real-world network, we should take into account that while the number of nodes soars, other structural changes (i.e. densification, increase of average degree) affect the network so that network diameter might even plummet as the network grows [31].

2.1.4 Centrality

Centrality is generally described as a set of different measures designed to determine how well-connected or influential each node is in the network [16]. These measures are usually defined through connectivity properties of the nodes in question.

One measure, *degree centrality*, uses degree of each node [18]. This is rather straightforward to calculate and understand, but it is also a very local measure that only takes into account the immediate vicinity of each node. *Closeness centrality*, on the other hand, uses reciprocal of the average of all shortest path lengths of one node to all other nodes as a measure of centrality for that node. A node that is on average closer to all other nodes is, therefore, more central according to this measure [18]. Another measure is to count how many shortest paths from all pairs of nodes pass through a certain node. This measure is called *betweenness centrality* [32]. Figure 1 compares betweenness and closeness centralities in context of one simple real-world network.

It is also possible to devise centrality measures not for nodes, but for links. This family of measures are called *edge centrality*. For instance, one can easily extend betweenness centrality to create a measure of *edge betweenness centrality*, which is based on the number of shortest paths that pass through each edge [33].

2.1.5 Percolation theory

Percolation theory is a sub-field of statistical physics that describes formation and behaviour of connected clusters in random systems [34]. Although percolation theory's origin is rooted in the study of lattices and random systems. However, the concept and theory are also useful in the study of connectivity of empirical networks such as the transportation networks studied here.

Percolation theory can be used to model a variety of real-world phenomena. An example often used to illustrate this is the effects of vaccination on spreading of a disease. Each person can transmit the disease through a mean of contact, e.g. sexually, to other people. We can create a network where people are represented as nodes and their contact as links. A person vaccinated cannot get ill and cannot transmit the disease to other people, which can be represented by removing a node from the network. Vaccination of a person, therefore; not only makes the vaccinated person immune, but also reduces the risk of infection for other people in contact with them. This makes it possible to prevent the spread of a disease by only vaccinating a minority of people [18]. Another method of reducing the risk of an epidemic is to cut links between nodes, by e.g. quarantining, in effect "compartmentalising" the population so that a disease breaking out in one community cannot spread to the whole population.

It is that study of percolation is in nature a study of connectivity of networks. Effects of non-functioning or removed nodes or links can be measured in this manner with tools established before such as component sizes and path lengths.

An example frequently used to mathematically describe the subject is a lattice where we "occupy" each node (also known in this context as *sites*) with probability p. If we consider neighbouring occupied nodes as connected to each other, it can be shown that with a small p, the size of largest component is small —or zero for an infinitely large lattice. With p much smaller than p_c, the component sizes can be shown to have an exponential distribution [29].

As we increase p the size of all components grow. When p approaches a critical value p_c we see that small connected components merge and coalesce leading to formation of an infinitely large *percolating cluster* (also called a *spanning cluster*) that extends through the infinitely large lattice. Formation of a percolating cluster in this fashion is sometimes called a *percolation transition*. At $p = p_c$ the component sizes distribution is power law [29]. As p increases past the critical threshold, sizes of small components decrease very fast as bigger components are more likely to get attached to the largest component with occupation of more nodes.

The numerical value of p_c depends only on the local structure of the lattice. As with many other systems showing critical phenomena, certain universal constants that govern the evolution of the system. These universal constants are completely independent of the actual dynamics of the system and local structure of the lattice, and only a function of number of dimensions of the lattice [35]. Figure 4 shows an example of this phase change and Figure 5 shows the change in mean cluster size, mean size of small clusters and probability of an occupied node belonging to the largest component or the second largest component in a large 2-dimensional lattice. This approach can be generalised to study robustness of networks, which is introduced in Section 2.1.6.

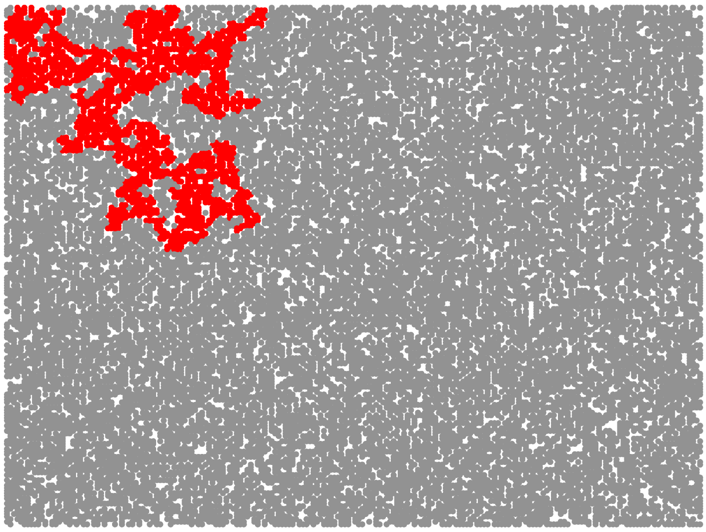
A similar system can be devised by creating a full lattice and connecting neighbouring nodes with links (or *bonds*) with probability p. This system also shows a phase transition around a certain threshold for p, but it should be noted that although the numeric value for this *bond percolation threshold* is also only dependent on the local structure of the lattice, it is almost never the same as the *site percolation threshold* discussed above.

2.1.6 Error and attack tolerance

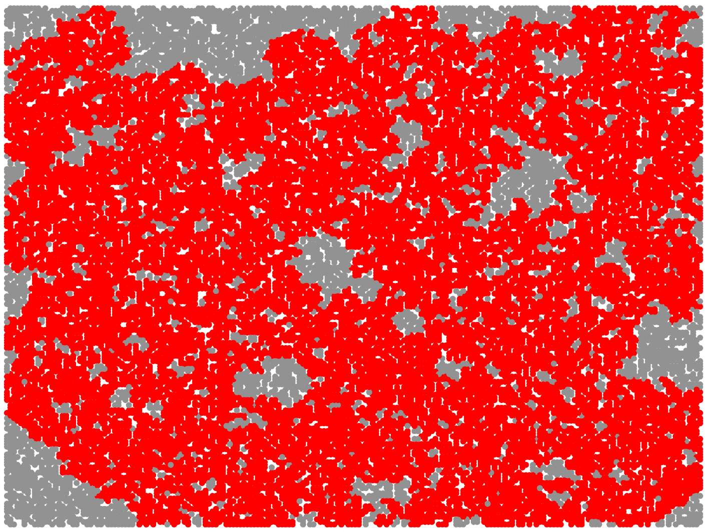
Many natural phenomena show a very high degree of resilience against fragmentation or malfunction when faced with problems that lead to the removal of entities or connections between them. Biological systems maintain phenotypic stability when faced with random perturbation from the environment or genetic variations [36]. At another level, more complex forms of living organisms consisting of many smaller biological subsystems are usually able to handle losing many of those subsystems regularly. Many pieces of evolved infrastructure, such as the internet, show similar resilience toward regularly losing subsystems without experiencing total failure. Successful modelling of robustness in context of complex networks, their ability to withstand fragmentation or other impediments in face of random failure (errors) or targeted attacks, is a matter of great importance in many disciplines.

A common model for studying this behaviour is the percolation model, borrowed from statistical physics, in which the properties of connected components are studied. We previously discussed percolation theory on Section 2.1.5. Implementation of ideas borrowed from percolation on complex networks focuses on measuring network fragmentation or various other network statistics after successively removing nodes or, depending on the problem at hand, links [35].

An application of this method for networks can be seen in the article titled "Error

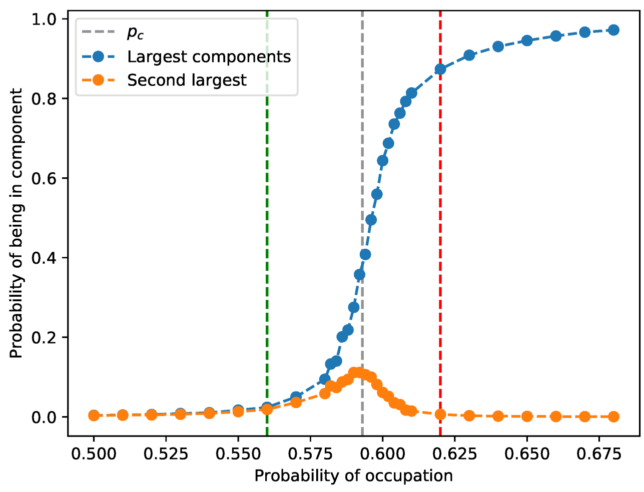


(a)

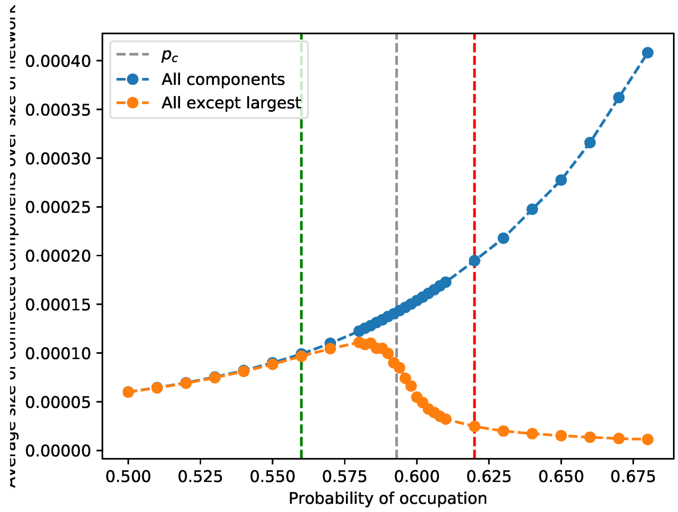


(b)

Figure 4: A 2D lattice of size 200×200 with largest connected component coloured red. Panel a has a occupation probability of $p = 0.56$ and Panel b has a probability of $p = 0.62$. Critical occupation probability p_c for a large 2D lattice is $p_c \approx 0.593$. Note how the largest connected component suddenly grows to a size comparable to that of the whole lattice.



(a)



(b)

Figure 5: The probability of occupation and mean size of components measured as a function of occupation probability. An ensemble of 50 lattices of size 500×500 was constructed for occupation probabilities between 0.50 and 0.69. Panel a shows the probability of a uniformly random node belonging to the largest connected component and the second largest component and Panel b measures the mean size of all components and the mean size of all components except the largest. For occupation probabilities larger than the critical occupation probability $p_c \approx 0.593$, shown by the grey vertical line, the largest component starts to dominate the network at the cost of smaller connected components, while for occupation probabilities before the critical values, both seem to be growing together. This shows a clear change in behaviour around the critical value. The occupation probabilities corresponding to the two lattices from Figure 4 are marked with green and red vertical lines.

and attack tolerance of complex networks" [37] wherein the tolerance of randomly generated Erdős-Rényi networks are compared to scale-free random networks generated through the model proposed by Albert-László Barabási and Réka Albert and two real-world networks (WWW and the internet) reported to be scale-free by the authors. Results show that the special stability of scale-free networks towards random failure is not just an effect of increased redundancy, but a feature of degree inhomogeneity of scale-free networks. They also indicated that while scale-free networks are much more robust to random failure compared to random networks, they are also much more vulnerable to targeted attacks where nodes with higher degrees are specifically targeted.

2.2 Temporal networks

The above notions and theory of connectivity were developed in the context of static networks that do not change in time. Clearly many systems, such as the public transportation network that is the topic of this thesis, are not static in nature. Even though one can draw a static map of public transportation, the buses, trains and other vehicles in these networks follow timetables and already by experience most of us know that ignoring this temporal aspect would be catastrophic when travelling in a complicated public transportation system. Therefore, the notions of connectivity developed for static networks and the theory that accompanies them need to be generalized to networks that are embedded in time.

Often in the past, phenomena with effects or relations with a dynamic nature were modelled using static networks by aggregating the interactions into an adjacency matrix [6]. This method makes studying such systems simple and compatible with the methods already established for static networks. However, in the process, we lose information regarding the system dynamics.

One compromise is to split the duration of study in time windows and aggregate all the interactions between each pair of nodes over the window duration, e.g. two nodes are connected if there is at least one interaction in the time window between the two. This way we still have to work with a few static networks, but to compute certain features of the network with the time dimension in mind, one has to change the way they perform certain computations and look at the model as a multilayer network where the network for each time window is represented as a layer [6, 38]. Various tools exist to work with multilayer networks e.g. extensions of community detection algorithms enables us to study the evolution of communities across time with this time-window approach [38].

Another solution is to model the dynamic system as a temporal network [6]. Temporal networks are networks where links are active only at certain points in time or specific time periods. A more formal definition of temporal networks would be as a set of nodes V and a set of events C.

Two broad classes of temporal networks have been defined in the literature [6]. One class is where each event is only momentarily active. This means that each event, each member of C, is a 3-tuple consisting of two nodes and one timestamp. An example of this type of systems is a network of email communication, where delivery of an email is for all practical proposes happen instantly.

Another class of temporal networks have events that are active in intervals, in which case each event is a 4-tuple consisting of two nodes and two timestamps. An example of such system is a network of phone calls, where each call has a duration.

Many concepts of static networks do not have a clear temporal analogues, but analogues exist for some of the statistics described for static networks in Section 2.1 which we will review promptly. There are also some new concepts that only apply to temporal networks.

Some examples of phenomena that have been modelled as temporal networks include phone calls [39], email [40] and other forms of person-to-person communication [6], epidemics [41] and trade between commercial entities [42].

2.2.1 Distances, latencies, shortest paths and connectedness

Clearly, a simple geodesic distance, similar to the definition of shortest path length for static networks, is not applicable for the case of temporal networks since connection between two nodes is a function of time. Furthermore, the concept of path length that in static context was used as a proxy to study how fast information or effects propagate throughout the network is redundant as we know exact timings of the events and time between events as a separate statistic and decouple it from the number of links between two nodes. We will next define vocabulary that will be useful during the course of this Thesis.

An effect, which is any process mediated through events on the temporal network, originating at time t_0 from a node i can reach node j at a time t only if there is a sequence of events that can be causally chained together; meaning that any event in the series should start after the end of the previous event and before the start of the next in the sequence that start after t_0 from i and end at some point before t on j. This sequence of events is the analogue of paths in the context of temporal networks. If we explore all such paths and minimize time t, then $t - t_0$ describes the *forward latency* from i to j at time t_0. In literature, terms temporal distance [43] and latency [6] are sometimes used to describe this value. There is also no reason for this value to be symmetric for a pair of nodes in general, even if the events themselves might be bidirectional. That is, the forward latency from node i to node j at time t_0 is not necessarily the same as it is from node j to node i at time t_0. In keeping with transportation nomenclature [44, 45] we will use “origin” to describe i and “destination” (or “terminal”) for j in this Thesis.

An important distinction of the definition of temporal events in the context of this Thesis is that in the cases we are interested in, events do not keep transmitting information between origin and destination throughout all the duration between their beginning and their end, but these are merely describing the delay between transmission from the origin to destination. To illustrate, an example of an event in this context is a bus trip between two consecutive stops: If a passenger is at the origin stop at any time $t < t_0$ where t_0 is the time the bus departs from the stop, they can be transported to the next stop and the transfer takes time $\Delta t = t - t_0$ until the arrival at the next stop. But if they arrive at any time $t' > t_0$, they won’t get a chance to board that bus as it has already departed. In this Thesis, *boarding time* at the origin is considered infinitesimally short. We will, therefore, use (vehicle) *departure* and *arrival*, borrowed terms from transportation literature, to more clearly denote start and end of an event [44]. This is in contrast to a definition of temporal network where an event is active from time t_0 to time t and can keep propagating information as in a phone call network [39]. Both these definitions, and many more subtly different ones, are used in the literature depending on the phenomenon being studied so one needs to be careful about applying methods and statistics devised for one definition to another without paying attention to the compatibility of the method or statistic.

As the network is not static, the forward latency between every pair is constantly changing. A simple average latency, without some special considerations, e.g. not taking into account time boundaries of our observation, might lead to a distorted view of the network [43]. Since we can only numerically work with temporally finite networks, latencies in the times close to the start or end of the observation period might not reflect the phenomena as there might not be enough time to complete a chain of events to some destinations. One way to mitigate this issue is to repeat the events from the beginning of the observation period after the end of the observation period [43]. This, however, might not be necessary if values of latencies are much shorter than the observation period and we are not interested in the state of the system toward the end of observation period.

The analogue to the concept of shortest path can also be defined as a path that minimises a certain cost function or a set of cost functions. A set of paths that describe the best-case trade-off between cost functions is called a *Pareto frontier* and each path in the set is called a *Pareto-optimal path* [46]. It should be noted that as the paths themselves vary as a function of time, so will any cost function calculated based on them.

2.2.2 Reachability

One of the useful statistics and an indicator of connectivity of a temporal network is that how many nodes can be reached at a certain point in time t_0 in the observation period $[t_0, T]$ from a certain origin i. This set of possible destinations is called the *set of influence* of i [6]. A simple average of the fraction of nodes that fall within an origin's set of influence over all possible origins determines what is called the *reachability ratio* [6]. One can also find the subset of all possible origins that can reach a certain destination in a time period and derive a similar measure but backwards in time, finding possible causes of an effect at a node i in a certain point of time t_0. This set of nodes have been dubbed *source set* of i and the cardinality of that set, *source count* of i [6], and usually algorithms that calculate one can be adopted to find the other [6].

The notion of light cones has been used to denote the set of sources and influences of a node throughout time, which resonates well with the cause and effect line of reasoning used frequently in this sections [6]. Simply put, since no causal influence or information can propagate faster in space than the speed of light, an event can only be causing or affected by another event if the other event happened close enough in terms of space or far enough in terms of time that a theoretical photon travelling at the speed of light could have reached one from the other. When plotting the space-time with time in one axis and space on other axes, the location and time of possible causes or effects of an event happening in a certain point can be described as a double cone with apices on location and time of that event [47]. The speed of light here is a theoretical maximum on the propagation of information in space, which is similar to the idea of the forward and backward latencies on a temporal network.

2.2.3 Centrality

For some static network centrality measures, it is easy to devise analogues in the context of temporal networks. Shortest path has a clear temporal analogue as defined in section 2.2.1 and degree of a node can be replaced with number of activated links incident to that node [6] or some other similar definition. With these measures, one only needs to take into account that shortest paths and many other properties in temporal networks are defined as a function of time.

A *closeness centrality* analogue, similar to its static definition in section 2.1.4, can be constructed from average (or sum) of latencies of a certain node i to all other nodes [43, 6]. It is worth noting that closeness centrality is also a function of time and subject to the conditions described for measuring latencies in the previous section.

Another interesting and non-trivial case is the definition of a betweenness centrality analogue. As with the static analogue, it is the fraction of shortest paths that pass through a certain node in the network. The complication arises from the fact that the shortest path between two nodes changes throughout the observation period. A straightforward way of translating betweenness centrality to temporal networks is to add a dependence on time and track what fraction of shortest paths starting at time t from every node pass through focal node i as a measure of centrality of node i at time t [6]. It is semantically important to note that although those paths start at the same time, they don't necessarily pass through i at a single point in time.

One can easily see a way to translate these time-dependent measures to one that does not point to a specific point in time by averaging it over a period of time or sampling shortest paths within a time window, as long as they take into account the problems related to the boundaries of observation period as described in Section 2.2.1.

2.2.4 Error and attack tolerance

As with static networks, temporal networks can be subject to many forms of attacks and failures and their conditions observed. A simple evolution from the static approach would lead us to use measures of connectedness of the network, but there are opportunities to use novel measures of performance that are not available for static networks. As with the static networks, one can also mix in data not directly embedded in the network, e.g. spatial information, to come up with domain-specific measures of performance and attack strategies [48]. Specifically, we can take into account the new temporal dimension of the spread of an effect, latencies, as a basis for measures of performance.

3 Material and Methods

In this chapter, we build on the knowledge of complex networks, and especially temporal networks, to provide a model of public transportation networks, perform attacks and random removal (errors) of routes based on different measures and measure the effectiveness of the attacks or errors based on the increase in the travel times. The effectiveness of methods of attack of routes in increasing the travel, compared to the effectiveness of random removal, time can show whether an attack method (or the statistic used to devise that attack method) is significantly better than random at distinguishing more important routes or not.

3.1 Modelling of public transportation networks as temporal networks

To model a public transportation network, we consider each public transport stop as a node and each *connection*, e.g. a bus moving from stop A (or connection origin) to the successive stop B (or connection destination), as an edge or event with connection duration, the time it takes for the transportation medium to move from origin to destination, as an edge or event property. More specifically, each event is a one-way connection from an origin stop to a destination stop starting at departure time from the origin and ending at arrival time to the destination.

To take into account the ability to walk between stops and therefore make short-cuts and transfers, there are events which start at each stop every time a transportation medium arrives at any certain stop, originating at that stop to all nearby stops with a duration according to spatial distances. As we will see later, there is no need to take into account all possible events of this type beforehand to compute travel times and reachability as we can dynamically generate them when calculating travel times.

3.1.1 Sampling origins

The Connection Scan Algorithm, similar to Dijkstra's algorithm for shortest paths on static graphs, can measure distance (or travel time in case of public transportation networks) from a single node to all other nodes in the network. When computing the travel time distribution in more extensive public transportation networks, computing travel times from all stops to all others might not be feasible, or even desired, considering computation time. The simplest way to get around this issue is to sample the origin nodes. To make it easier to compare changes in travel times (e.g. as in a robustness analysis described in section 3.3) we re-used the same sample of origins in all scenarios.

3.2 The implementation

3.2.1 Data source

Original data we used was in form of static General Transit Feed Specification (GTFS) format [49] and pre-processed SQLite database [50] through DeCoNet Public Transport Network Data Repository [51, 52]. We elected to use timetables from a normal weekday (a Monday) for all the cities in this report. Some cities were omitted from the experiment due to not being fully compatible with the requirements set above or having a very low number of routes, leaving a total of 20 cities listed in table 1.

The list of input files, generated from GTFS files provided by the public transportation companies and agencies are as follows:

- A list of stops, consisting of a stop ID, a latitude and a longitude.
- A list of connections, generated from the timetable. Each entry indicates connection origin stop, connection destination stop, departure time, arrival time and a route ID.
- A pre-generated list of pseudo-connections, or stops that are closer than the 1-kilometre walking distance limit to each other. Each entity consists of an origin stop, a destination stop and a distance in metres.

Furthermore, in order to perform certain attack strategies, e.g. removing routes based on their capacity or frequency, we created a list where each entity consists of a route ID and the value of statistic we intend to use to attack the network, e.g. nominal capacity of that route.

In order to extract routes are part of metro service in each city, we looked at the route type information in GTFS files. Since some cities used original standard type numbering scheme and some used Extended GTFS Route Types [53], we came up with a list of all valid metro-related codes: 1 (original GTFS code for metro), 100 (Railway Service), 401 (Metro Service), 402 (Underground Service), 500 (Metro Service) and 600 (Underground Service).

3.2.2 Connection Scan Algorithm (CSA)

We use the Connection Scan Algorithm (CSA) [54] throughout this Thesis to find the shortest paths and latencies from one node to any other on the network. Connection scan algorithm scans through the list of events on the temporal network, called *connections* in the context of transportation networks. It takes into account the possibility to move to other nodes on foot at any point in time, by generating sets of events called *pseudo-connections* whenever their existence might be helpful in decreasing travel time, and thus calculates shortest paths to all other nodes from an origin node departing at a certain point in time. Each node of the network can also be called a “stop” or a “station” to keep with the nomenclature borrowed from transportation literature [44, 45].

The connection scan algorithm works by first putting all the connections in a minimum priority queue, with the priority assigned according to the departure time of each connection and setting the initial value of *reaching time* for each stop to infinity. The reaching time of origin node was set to the desired start time. Then at each step, the lowest priority element from the queue is popped and it is checked if taking that connection is causally possible, i.e. if the reaching time of the origin of that connection is earlier than the departure time of that connection. If taking the connection is possible, it is then checked if taking this connection improves the reaching time of the connection's destination by comparing the arrival time of the connection to the current reaching time of the destination. If this is true, the reaching time corresponding to that connection's destination is updated and all the possible pseudo-connections originating from that destination are inserted into the priority queue, with their departure time set to the new reaching time of the updated stop and their arrival time calculated using the time it takes to walk between the two stops. This event of finding a better reaching time to a stop is known as a *relaxation event*.

The final reaching times are the forward latencies from the origin stop to all other nodes. With a minor modification, we can also assemble the shortest paths to any destination: when we update the reaching time for a stop, we also record the event that leads to that update for the corresponding stops. With this, we can construct an event chain, a path, for each final destination all the way up to the origin recursively by looking at the last event that leads to a stop and chaining that with the shortest path of the origin of that last event. We can, therefore, calculate any cost function based on connections and time on each temporal shortest path.

The implementation of Connection Scan Algorithm produced for this Thesis follows the original paper quite closely [54]. For the proposes of this Thesis, we limit maximum total travel time to 5 hours and destinations that are not reached within this time are deemed unreachable. This reduces running time of the program considerably. We also limited the pseudo-connections to a distance of one kilometre and the walking speed to 1.4 meters per second, similar to the reported preferred walking speed consistent across different groups of people [55]. Algorithm 1 demonstrates the implementation of CSA for this Thesis in more details.

To perform robustness analysis, we first select a set of routes to remove according to a attack scenario (the line of reasoning behind this is explored in more details in Section 3.3). Then we scan through the list of connections while skipping those connections that are part of the removed routes and add the rest to a minimum priority queue where departure time determines the priority. For each origin from the sample, the implementation takes a copy of this priority queue and runs CSA as described in Algorithm 1. As the output, the implementation yields an associative array of stops and the last connection which gives the best arrival time to that stop as a function of a certain origin and departure time. This associative array can be used to form the shortest path between origin and a final destination.

In order to compute the delays, the same procedure is repeated without removing any routes and travel times between each specific origin and destination are compared. This set of "normal" travel times can also be used to see how the network behaves

Algorithm 1: Pseudo-code of our implementation of CSA; note that there is no need to keep track of arrival times separately as *reachability* members already has an arrival time attached to them but here we elected to use a separate variable to improve readability.

Data: a minimum priority list q of connections with departure time as priority
Data: an origin stop *root* and a starting time t_0
Data: an associative array *pseudoconnections* from each stop to a list of $(stop, walking\_time)$
Data: a set *stops* containing all the stops
Result: associative array *reachability* from *stop* to *last connection*

$arrival\_time \leftarrow \{\}$;
$reachability \leftarrow \{\}$;
for $stop \in stops$ **do**
    $arrival\_time[stop] \leftarrow \infty$
$arrival\_time[root] \leftarrow t_0$;
for $(other\_stop, walk\_time) \in pseudoconnections[root]$ **do**
    $pseudoconn \leftarrow (root, other\_stop, t_0, t_0 + walk\_time, \ldots)$;
    insert *pseudoconn* into q;
while *q is not empty* **do**
    pop the connection $conn = (origin, destination, departure, arrival, \ldots)$ with lowest priority from q;
    if *departure > maximum time limit* **then**
        return *reachability*;
    else if $departure \leq arrival\_time[origin]$ *and* $arrival < arrival\_time[destination]$ **then**
        $reachability[destination] \leftarrow conn$; `/* a "relaxation event" */`
        $arrival\_time[destination] \leftarrow arrival$;
        for $(other\_stop, walk\_time) \in pseudoconnections[destination]$ **do**
            $pseudoconn \leftarrow (destination, other\_stop, arrival, arrival + walk\_time, \ldots)$;
            insert *pseudoconn* into q;
return *reachability*;

unaltered, e.g. to find which routes are more likely to be used in a time-optimised travel between two randomly selected stops.

3.3 Analysis of the robustness of the network

Study of robustness of networks gives insight into the connectivity of the network. Comparing the delay distributions and access histograms of different strategies to uniform random removal (error) offers insight about the importance of certain statistics to topological structure of the network as well as about vulnerabilities of the network. If a method of attack devised using a certain statistic, e.g. based on a centrality measure, incur higher delays earlier than random removal, we can deduce that the statistic behind the attack method might have more significance in the structure of network compared to a statistic that performs similarly to uniform random removal.

Because in most forms of public transportation networks the mode of transportation operates in routes, a set of connections that correspond to bus lines, metro lines, tram lines etc., simply removing a stop or connection between two stops is highly unrealistic and exploring the results of such an attack does not offer much in terms of intuition vis-à-vis inherent network structure. To this end, we decided to focus attack scenarios on removing whole routes. We devised a series of experiments where we remove routes with different strategies to study how delays of all origin-destination pairs are distributed and plotting access histograms which show what fraction of origins have access to what fraction of the network within a time limit.

Since some failures and attack strategies deal with randomly removing routes, there is always a chance that results of a single attack trial is not reliable enough to base conclusions upon. Therefore, for random removal strategies instead of delay distributions we use average delay distributions where we use the average of different delay distribution functions obtained through applications of the attack.

The popularity of routes in “normal” shortest paths, shortest paths on the unaltered network, given a large enough sample of origins and departure times, can be used as a measure of “route betweenness centrality” given the similarity between that definition of edge betweenness centrality in static or temporal networks as defined in section 2.2.3.

3.3.1 Different attack strategies

The first and one of the most simple attack (or failure) strategies consists of removing a number of routes selected uniformly at random. We call this method **random removal**, **random failure** or simply **error**. The resulting delay distribution (averaged across many applications with the same number of routes removed) shows how the network connectivity depends on random routes and it can also be used as a baseline to which other, more targeted attack strategies can be compared to. Resulting accessibility histogram can offer insight on the state of the network after the attack, e.g. what fraction of sample origins get totally or partially disconnected from the network after the attack.

Another attack strategy consists of **removing a mode of transportation** completely and comparing that to the normal network. This is especially useful for determining the importance of auxiliary modes of transportation and their effect on accessibility in cities that have more than one dominant mode of transportation.

It is also plausible to remove routes based on their **nominal capacities**. As public transportation networks are expensive to maintain, routes with high capacity might be of great importance to the structural integrity of the system.

A common theme in studies of robustness in static complex networks is to come up with attack strategies based on different properties of nodes or links, e.g. degree, local clustering coefficient or betweenness centrality. This gets more complicated in our case because we are focusing on the failure of routes (as opposed to stops or connections, our equivalent of nodes and links) and also because many of the well-defined properties of static networks currently do not translate into a direct analogue for temporal networks. It is possible, however, to derive statistics for entities based on an aggregated, static network and use those for determining attack strategies. To demonstrate this, we constructed a static route-route network where routes are nodes of the network and two routes share a link if they share at least one stop. We then calculated various centrality measures for the route and used that information to determine the order of removing routes in the temporal network. We used **(static) betweenness centrality**, **flow betweenness centrality**, **Katz centrality** and **eigenvector betweenness centrality** of the route-route network as attack measures.

3.3.2 Quantifying effects of attack and error on travel time

An easy way to quantify the effect of a method of attack or error is to use the average delay as a function of removed fraction of routes. Since the average delay as a function of removed routes have the same value for all attack methods when the fraction of removed routes is 0 or 1, the total area under the curve of the function would be higher for an attack method that deals with an effective blow to the network faster. Furthermore, we can normalise the average delays by dividing them to the average travel time of the city, as an analogue for size of the city in a temporal frame of mind, so that we can compare values between different cities. Hereafter we will call the normalised version of this quantity "effectiveness of the attack or error". Effectiveness of attack is a dimensionless quantity.

3.3.3 Parallelising robustness analysis in a distributed computing environment

As many of our methods rely on running the same travel time calculations multiple times just with different sets of randomly selected routes removed, it is inherently an easy problem to parallelise. The calculations were performed using computer resources within the Aalto University School of Science "Science-IT" project. Each of the instances of the analysis was run as a separate process, commonly on different servers, the results streamed and saved as different files on the disk and later aggregated through another program that produced the plots. Each attack scenario takes between one to five minutes for a sample size for 200 origins depending on the number of connections for that city, number of relaxation events (see 3.2.2) and average number of pseudo-connections.

3.4 Distribution of accessibility in the network

Accessibility in public transportation networks, a measure of "ease" of reaching potential destinations from an origin, has been considered essential to ensure equality [56] and lower levels of it considered a transport disadvantage and a barrier to equal opportunities for everyone [57] where low accessibility cutting access to education, employment and healthcare with disproportionately higher impact on low-income families due to their lack of access to other, private means of transportation.

In a broader sense, lack of physical access is only one aspect of social exclusion, which is a process that through a set of constraints, limits the ability of certain groups or individuals to participate in normal activities of the society. In many cases, this definition is immediately linked to economically challenged groups, but it can affect other groups as well [58]. A hypothetical situation where different levels of access are available to different neighbourhoods in cities where minorities live in localised neighbourhoods, or in urban versus suburban areas, is not out of the question. Although we did not pursue any specific studies in this direction.

We simplified the measure of cost, ease of access, as the time spent travelling from an origin to each destination. Specifically, we aim to see how many destinations can be reached from each member of a random subset of origins in a set amounts of time. This way, given enough sample origins, we can plot the distribution of accessible destinations for a random origin.

3.5 Using the implementation

The analysis pipeline is designed to be easy to take apart and use in different contexts. Different steps of the pipeline exchange information through tab-separated-values text files. All binaries and scripts log to standard error and write out the results to standard output or directory structure defined as command line options. Internally, data is represented in C++11 Standard Template Library containers and a few simple structures to represent domain-specific data, e.g. origin-destination pairs or connections. A few simple Python scripts are used to produce vector graphics using the Matplotlib plotting library [59].

Possible uses of the implementation are not limited to public transportation domain. The connection scan algorithm implementation (in `connection_scan.cpp` and on a higher level in `robustness.cpp`) can be used for finding shortest paths or propagation in any temporal network, given that one can prepare the input data in the correct form. In any other context, there is typically no need to use some apparatus similar to pseudo-connections. The raw output of the connection scan implementation, namely a list of origin, destination, departure and arrival time tuples then can be visualised using the code available in `travel_time_hist.cpp` to produce a histogram of travel times.

The output of the Connection Scan implementation can also be compared to another raw output (i.e. an attack scenario) using code available in `mean_delay_stops.cpp` to produce average delay time for each stop. That, in turn, can be used to produce figures similar to Figure 16. There is also a helper script (`plot_delay_heatmap.py`) to fetch the map tiles and produce the final visualisation.

Another possibility is to compare a "normal" travel times file to one or many trials of the same scenario to produce an average distribution of delays as in Figure 8. This can be achieved using code from `mean_delay_pdf_hist.cpp`.

A simple departure time distribution of the connections can be produced using the code in `connections_hist.cpp`.

Histogram implementation (in `histogram.h`) provides a general purpose set of tools for linear or logarithmic binning of any data type with defined comparison operators to double-precision floating-point. This is used in many of the other histogram-producing code in our implementation. The output, a list of the left edge, right edge and frequency for each bin, can be used as input to `plot_hist.py` or any other visualisation library.

The source codes for this report can be accessed in form of a Git repository branch through the URL https://version.aalto.fi/gitlab/badiea1/connection_scan/commits/thesis. For more detailed instructions on building the code and providing arguments to the binaries and the pipeline, check out the `makefile` at the root of the branch. The original code can be used under terms permitted by the MIT license.

4 Results

In this chapter, temporal network representations of public transportation networks of 20 cities are constructed and the attack strategies described in Section 3.3.1 are applied to each representation. The resulting delay distributions and accessibility histograms are then illustrated and effectiveness of each attack method on each city is calculated. A higher level interpretation of the results is presented in Chapter 5 Referencessec:discussion.

4.1 Basic properties

Table 1 shows some general statistics on transportation networks of cities used in the experiments. Distributions of departure times of connections for two of the cities are shown in Figure 6. The distributions generally show an approximately constant number of departures. Some cities show a slight upwards or downwards trend in the number of connections and some show spikes of increased departures at regular intervals. Figure A1 shows distribution of departure times for all the cities.

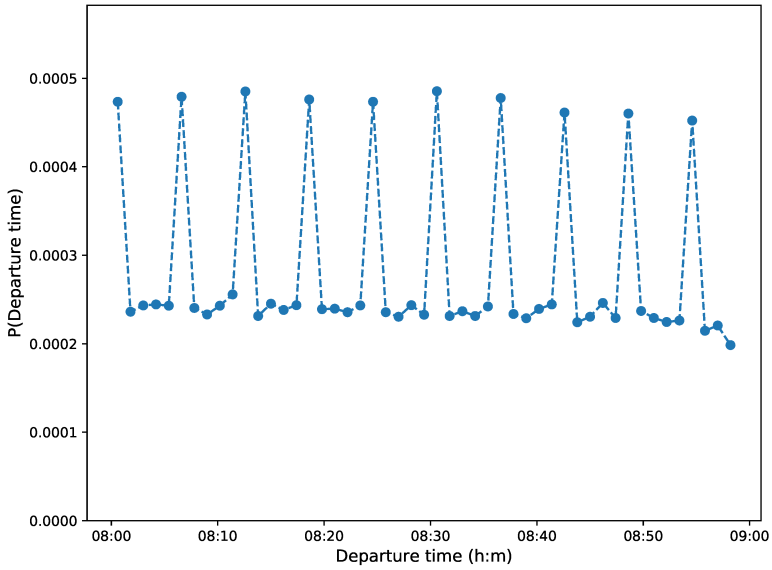
4.2 Travel time

We applied the algorithm described in Section 3.2.2 to the "normal" (unaltered) public transportation networks of 20 cities. These cities currently host a variety of public transportation modes: buses, trams, ferries, metro and different types of trains. We limited our studies to morning rush hour, departures between 8:00 and 9:00, and to a maximum duration of five hours.

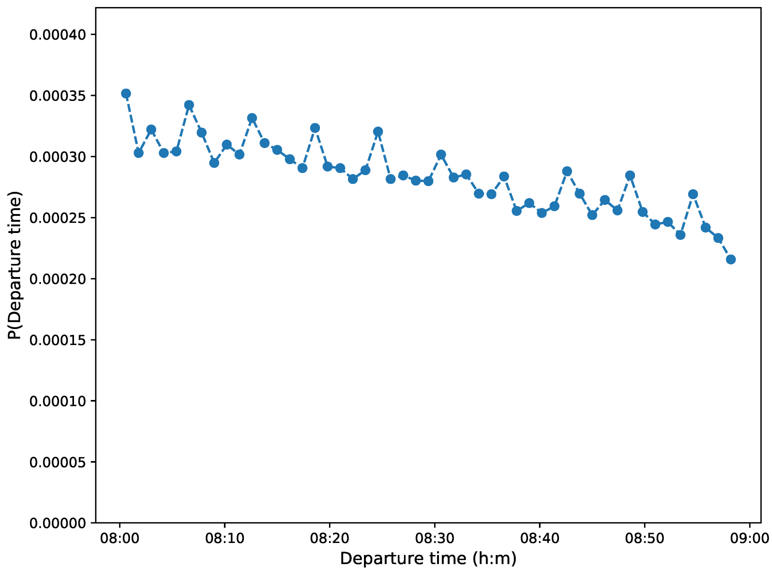
Figure 7 shows the probability density estimate for travel times of Helsinki and Adelaide estimated from travels from a sample of 200 origins to all destinations reachable in less than five hours. It has been reported that travel times in public transportation networks follow a log-normal distribution [60, 61] though we were unable to verify this through common normality testing methods such as Kolmogorov–Smirnov test, Kolmogorov–Smirnov test and Anderson–Darling test. These tests unanimously rejected the hypothesis that the samples were drawn from a log-normal distribution, but this might be due to the increased sensitivity of the tests as a result of the sheer number of samples which commonly numbered millions. Figure A2 shows the probability density estimate for travel times for all the cities in the experiment.

4.2.1 Route betweenness centrality

When computing normal (unaltered) travel times from every origin in our 200-origin sample to all destinations, we measure how many times each route was used to reach a destination in a shortest temporal path. As described in section 3.3, we use the probability of each route being used when travelling from a random origin to a random destination as a measure of "route betweenness centrality" and use it as a measure to devise attacks.

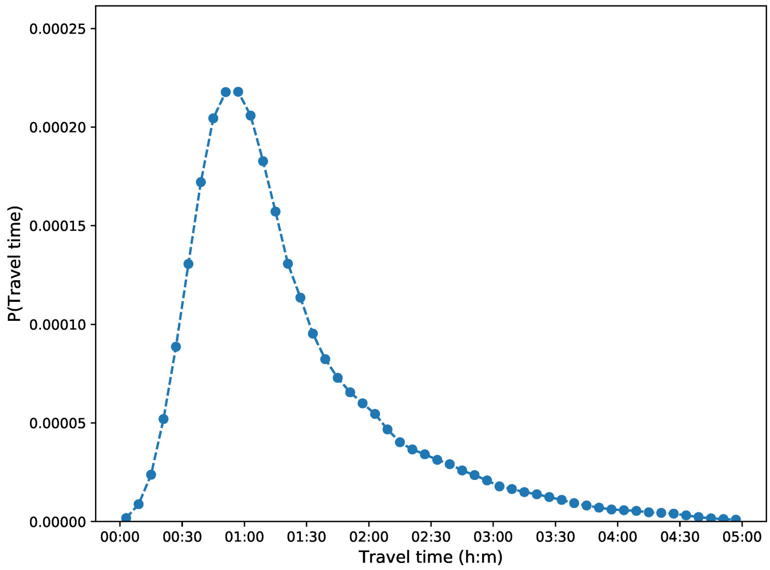


(a)

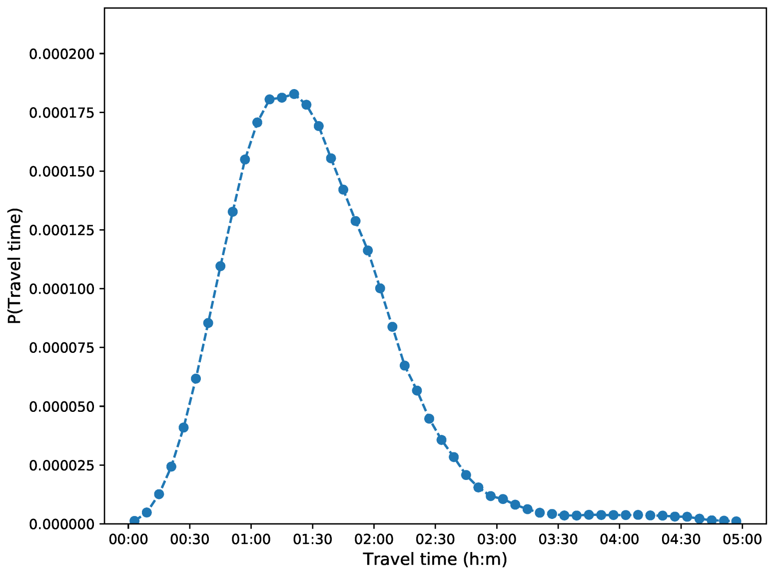


(b)

Figure 6: Probability density estimates for departure time of each connection in Helsinki (6a) and Adelaide (6b) and connections departing between 8:00 and 9:00.



(a)



(b)

Figure 7: Probability density estimates for travel times between all origin and destination pairs in Helsinki (7a) and Adelaide (7b). The estimates are based on sampling 200 origins and calculating the travel times to all reachable destinations. See section 4.2.

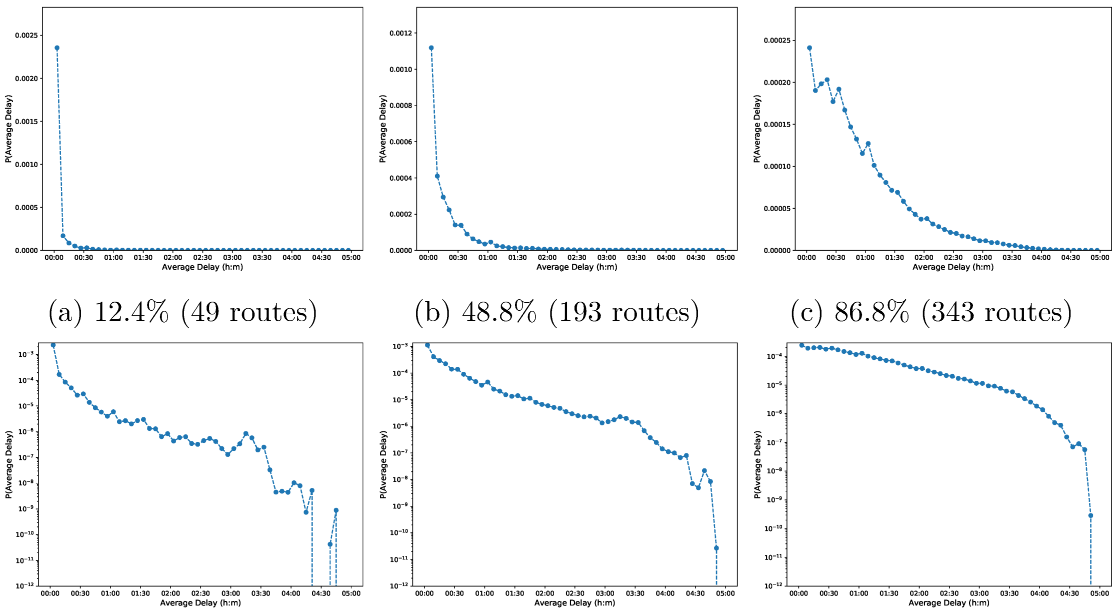
Table 1: General information on all the cities in the experiment. “Routes” denotes number of routes used at least once in any shortest path in the time period of the experiment. This filters out routes with no measurable effect in the period of observation. “Travel Time” is average travel time between sample of 200 origins and all destination measured in normal configuration without any attack or error.

City	Stops	Connections	Routes	Travel Time (m:s)
Adelaide	7 679	404 301	271	90:31
Athens	7 009	725 062	229	83:05
Belfast	1 918	122 693	82	78:40
Bordeaux	5 319	275 218	64	72:51
Brisbane	9 848	392 805	320	116:09
Canberra	2 821	124 683	99	93:48
Detroit	5 683	214 681	39	69:04
Dublin	4 620	610 966	129	75:26
Helsinki	7 099	686 457	395	81:42
Kuopio	552	32 122	30	55:55
Lisbon	7 098	526 151	129	92:37
Melbourne	19 649	1 139 771	395	119:11
Nantes	3 442	196 421	30	69:40
Palermo	2 176	226 215	550	57:58
Prague	5 491	704 843	416	98:03
Rennes	1 415	109 075	66	67:07
Sydney	24 410	1 759 775	593	124:18
Toulouse	4 999	224 516	94	89:56
Venice	1 876	118 563	320	121:24
Winnipeg	5 085	333 883	84	57:35

4.2.2 Static route–route network

In order to study route betweenness centrality in the temporal transportation network and its relationship to the static network equivalents, we constructed a route–route network by using each route as a node and having two nodes connected if they shared a stop. Through this route-route network, we calculated a variety of different measures of centrality for each route.

We found that none of the measures of centrality we tried, shortest-path betweenness centrality, Katz [62, 18] and eigenvector centrality [63, 18] and Current-Flow betweenness centrality [64], could be reliably correlated (in terms of rank-orders of the routes) with route betweenness centrality or nominal capacity of routes. The results of this correlation analysis is explored in Section 4.4.



(a) 12.4% (49 routes) (b) 48.8% (193 routes) (c) 86.8% (343 routes)

(d) 12.4% (49 routes) (e) 48.8% (193 routes) (f) 86.8% (343 routes)

Figure 8: Probability density estimates for average increase in travel time (i.e. average delays) due to random removal of a fraction of routes in Helsinki. Each estimate is built by averaging delays over 100 different trials of randomly selecting a set number of routes to remove. Panels a, b and c are in linear scale and Panels d, e and f have a logarithmic Y axis.

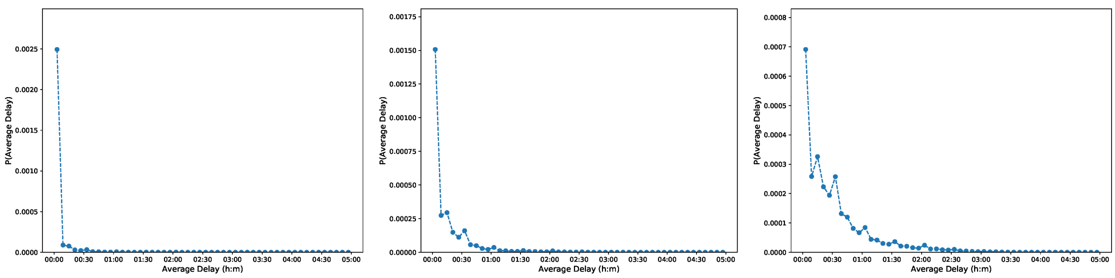
4.3 Error tolerance of transportation networks

With the assumption that the failure of the routes is a result of some random process, distributions of the average delays for each pair of stops in different public transportation networks are shown for Helsinki (Figures 8) and Adelaide (Figure 9). The distribution seems to be following an exponential distribution for a larger number of removed routes. The values of the distribution should not be trusted near the 5 hours travel time cut-off imposed on the implementation which limits the possibility of registering delays for pairs where after the attack travel time goes above five hours.

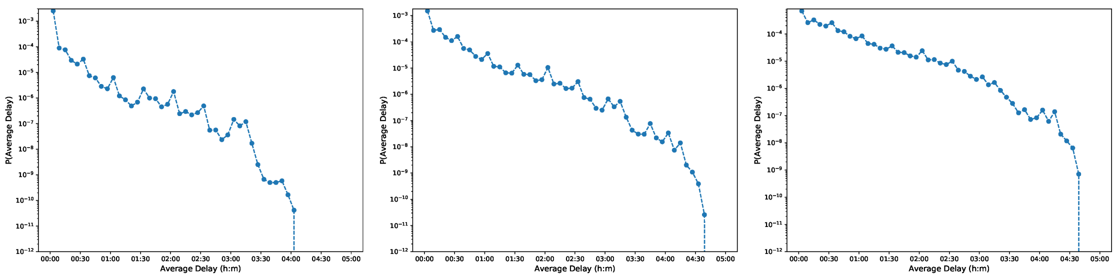
For each value of removed routes, we performed 100 different random removals of routes and travel time measurement. The average delays are then computed across the whole ensemble. Delay distributions seem to follow an exponential decline if taken into account that the final few points of the estimate are not to be trusted due to the 5 hours travel time cut-off imposed on the implementation which limits the possibility of longer delays more than shorter ones.

4.3.1 Attack tolerance of public transportation networks

As described in section 3.3.1, we elected to perform an attack strategy based on the **nominal capacity** of each route, one based on removing a mode of transportation,

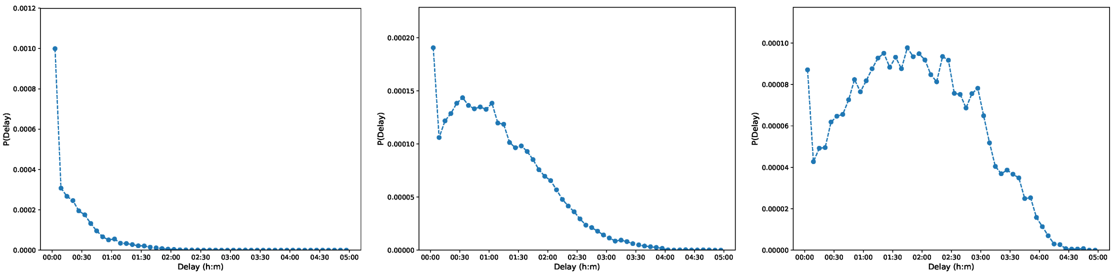


(a) 12.1% (33 routes) (b) 49.0% (133 routes) (c) 87.4% (237 routes)



(d) 12.1% (33 routes) (e) 49.0% (133 routes) (f) 87.4% (237 routes)

Figure 9: Probability density estimates for average increase in travel time (i.e. average delays) due to random removal of a fraction of routes in Adelaide. Each estimate is built by averaging delays over 100 different trials of randomly selecting a set number of routes to remove.



(a) 12.4% (49 routes) (b) 48.8% (193 routes) (c) 86.8% (343 routes)

Figure 10: Probability density estimates for average increase in travel time (i.e. average delays) due to removal of a fraction of routes based on their nominal capacity in Helsinki.

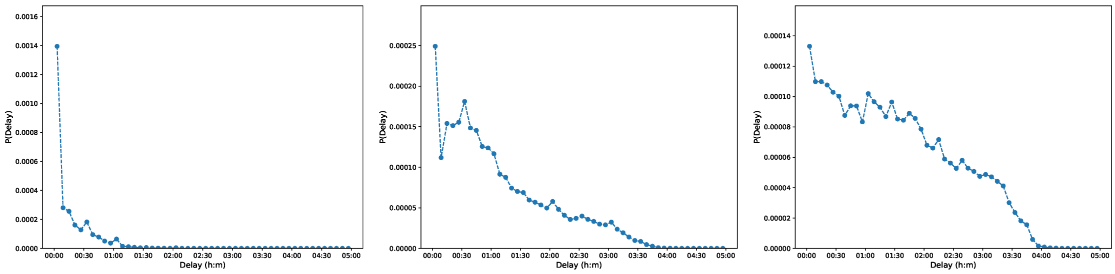
metro, four based on node centrality measures on aggregated, static route-route network (see section 4.2.2, namely **static betweenness centrality**, **flow betweenness centrality**, **Katz centrality** and **eigenvector centrality**). Additionally we used the probability of using the route in the shortest path between two random stops (**temporal route betweenness centrality** or **route use**) as another method of attack.

Nominal Capacity To calculate the nominal capacity of each route, we used seated passenger capacity based on that of most common variants of Helsinki public transportation vehicles [65], we only took measured capacity of routes that belonged to one the following classes: trams (44 seats), metro (124 seats), commuter trains (200 seats) and buses (35 seats). The number of seats per vehicle was then multiplied by the number of trips done on that route. The calculation was limited to Monday but not to any specific time window. The attack was performed by removing a fraction of routes, higher capacity routes first, and measuring shortest temporal paths.

Figures 10 (Helsinki) and 11 (Adelaide) show the effects of removing routes based on capacities on delays for typical examples out of 20 cities. Probability density of delays, compared to random removal (which roughly followed an exponential decline) has a completely different shape for higher fractions of routes removed. In contrast to random removal, whereas the number of removed routes grew, the shape of the distribution didn't change, the capacity-based attack with a lower number of removed routes has the same exponential distribution of delays as random, but gradually evolves into a bell-shaped distribution as the number of removed routes increases.

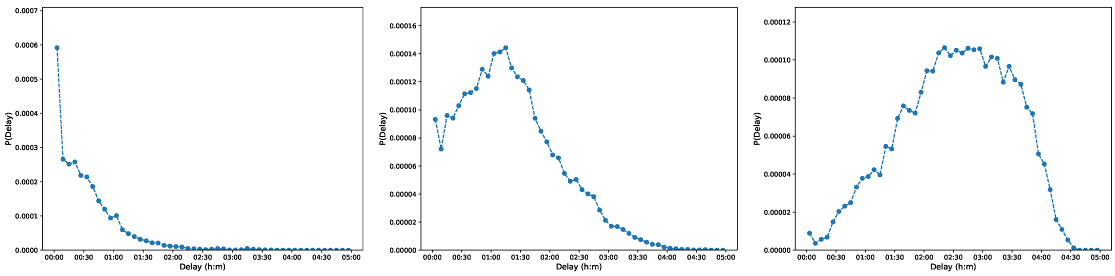
Route betweenness centrality To implement this method, we calculated route usage (or route betweenness centrality) as described in section 4.2.1 based on the probability of a route being used at least once in connections forming the shortest path from a random origin to a random destination.

Delays that are resulted from an attack based on route betweenness centrality, figures 12 and 13 being two examples, are distributed more or less similar to the



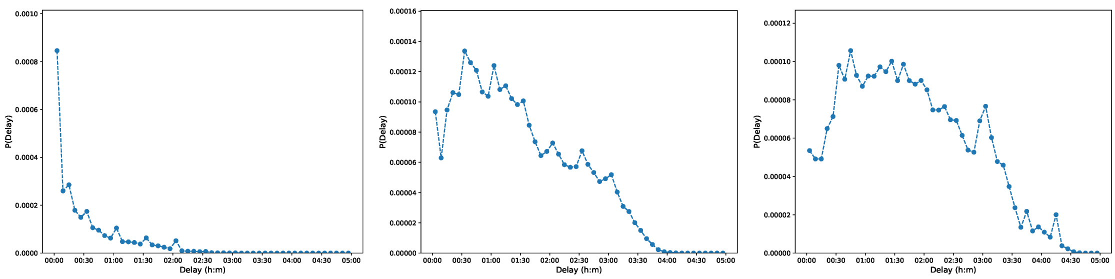
(a) 12.1% (33 routes) (b) 49.0% (133 routes) (c) 87.4% (237 routes)

Figure 11: Probability density estimates for average increase in travel time (i.e. average delays) due to removal of a fraction of routes based on their nominal capacity in Adelaide.



(a) 12.4% (49 routes) (b) 48.8% (193 routes) (c) 86.8% (343 routes)

Figure 12: Probability density estimates for average increase in travel time (i.e. average delays) due to removal of a fraction of routes based on their route betweenness centrality in Helsinki.



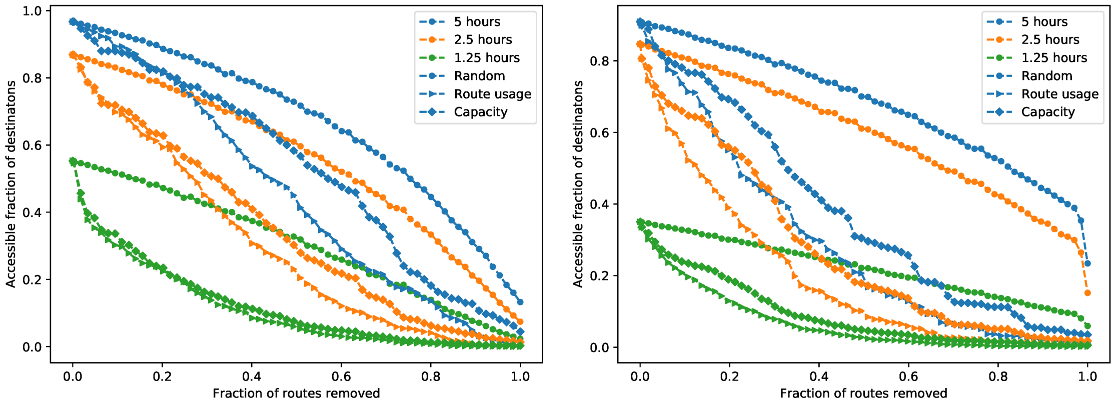
(a) 12.1% (33 routes) (b) 49.0% (133 routes) (c) 87.4% (237 routes)

Figure 13: Probability density estimates for average increase in travel time (i.e. average delays) due to removal of a fraction of routes based on their route betweenness centrality in Adelaide.

capacity based attack in the sense that they don't follow an exponential decrease, but have a bell-shaped distribution with a higher number of removed routes.

Figure 14 shows the changes to the level of access with different time limits and attack methods. Random removal of routes is by far the most inefficient in reducing access but the targeted attacks (based on nominal capacity and route betweenness centrality) perform relatively similar in this respect, meaning that they reduce average accessibility of origins much faster than uniform random removal. This trend is consistent across different values of time-out. Figure A3 shows changes to accessibility for all the cities in experiment. The same is true for almost all the cities except for Toulouse and Detroit where upon visual inspection the difference of attack methods and random removal on accessibility is not visible.

Figure 15 shows a similar trend in delays, with random removal and static methods being far less efficient (and relatively similar) and the two other methods performing relatively similarly. It should be noted that the travel times with a very large fraction of the routes removed are heavily influenced by walking times to nearby stops since travels longer than five hours were cut off and most stops, other than the ones accessible on foot, become inaccessible. Figure A4 shows similar delay plots for all the cities in the experiment. Here as well most of the cities show the same trend



(a) (b)

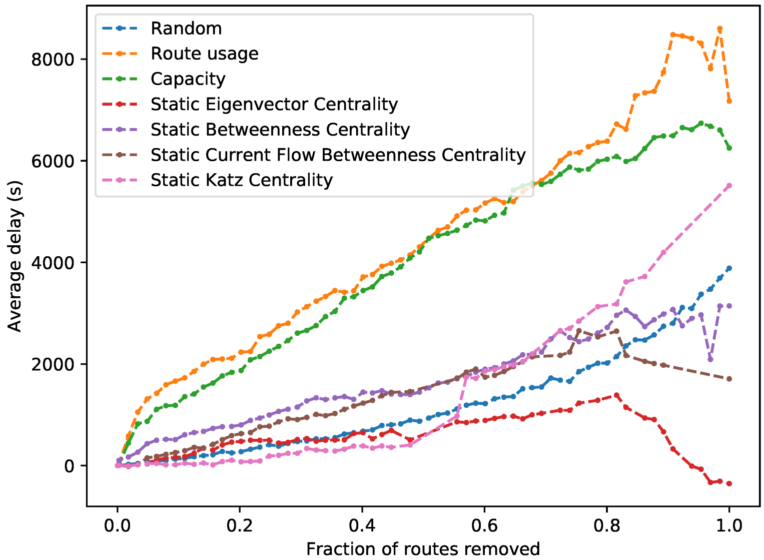
Figure 14: Decline in fraction of accessible destinations for different attack methods (marker shapes) and maximum acceptable times (colours) for Helsinki (a) and Adelaide (b). Standard error of fraction of accessible destinations for the case of random removal of routes was too small to show compared to the marker size.

as discussed but in a small number of cases like Palermo and Canberra the trend cannot be seen.

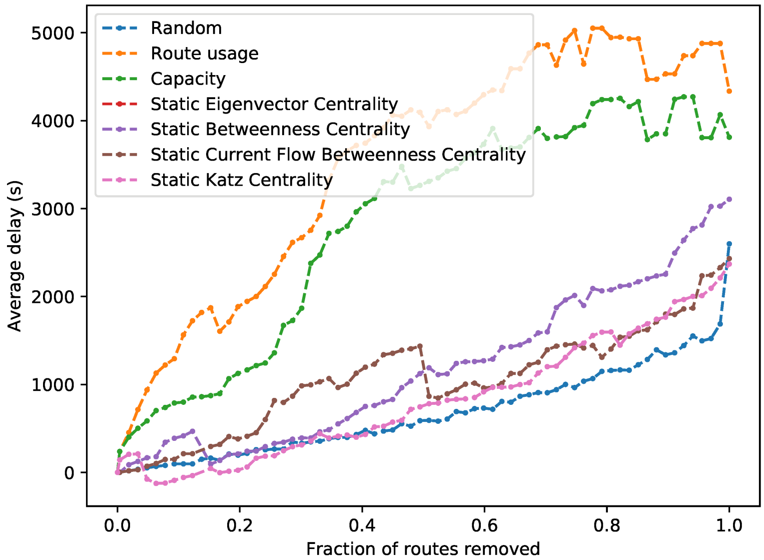
Removing a mode of transportation (metro) Only six cities out of the original 20 cities have metro lines reflected in their GTFS data, with the number of metro routes in each displayed in Table 2. Figure 16 shows geographic distribution of the delays suffered by removing metro from two cities. Prague (16b), due to the star-shaped spread of its three metro routes, show a wider spread of delays throughout all the city after removing metro, but in Helsinki (16a), where metro routes run along an east-west line on southern rim of the city, has only stops along that path affected by the attack.

Table 2: Information about number of metro routes in each city. “Metro Routes” denotes number of metro routes that have at least one connection on the given Monday. “Average Delay” shows average increase in travel time from random origin to a random destination as a result of removing all metro routes.

City	Metro Routes	Average Delay (s)
Athens	3	1 027.99
Helsinki	5	108.13
Lisbon	10	132.93
Prague	3	569.88
Rennes	1	357.45
Toulouse	2	1 092.21

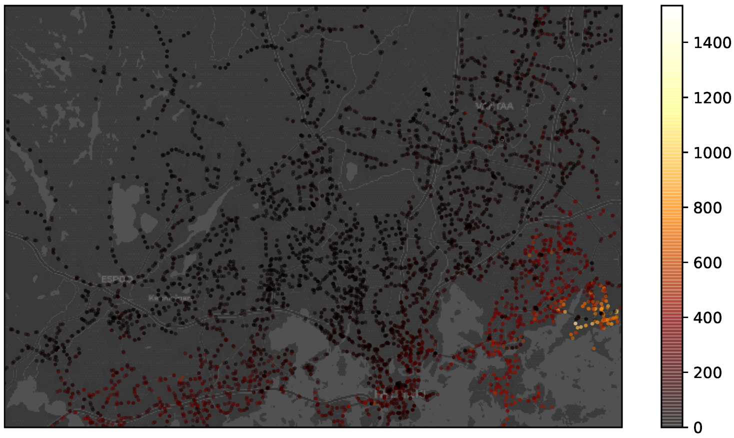


(a)

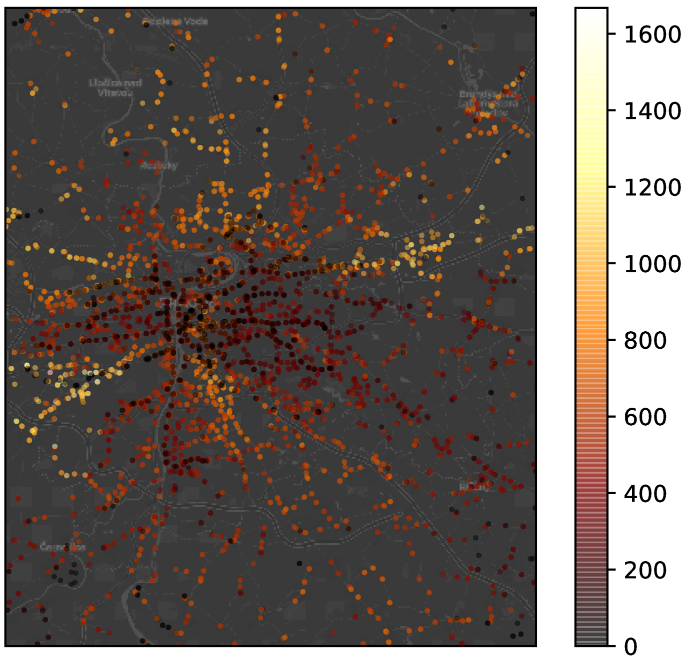


(b)

Figure 15: Average delay for different attack methods with different number of routes removed for Helsinki (a) and Adelaide (b). Standard error of delay for random removal of routes was too small to show compared to the marker size.



(a)



(b)

Figure 16: A heatmap of delays for based on the origins, after removing metro lines. Helsinki metro (figure a) is organized along east–west axis while Prague metro (figure b) has three lines, north-west–south-east line A, north-east–south-west line B and north–south line C.

Attacks based on route–route network centrality measures Finally we perform attacks based on four centrality statistics from an aggregated, static route-route network (see section 4.2.2). Figure 15 shows the average delay of two cities with the four attack methods based on centrality in route–route network, compared to random removal, temporal route betweenness centrality and nominal capacity. This results already suggest that these methods does not seem to perform any better than random removal of routes.

4.3.2 Effectiveness of attacks and error

Table 3 show the calculated effectiveness for all attacks and error methods as defined in section 3.3.2, except the method based on removing a single transportation medium, as that method does not produce an average delay as a function of removed routes as the number of removed routes is not variable.

Mean effectiveness values show that attack methods based on centrality in static route–route network do not increase delays any more efficient than random removal, but nominal capacity and route betweenness centrality perform much better in this respect.

Table 3: Effectiveness of each attack and error method on all cities based on the definition in section 3.3.2 shows how fast an attack method increases travel times when removing routes. The attack method based on removing a transportation medium is not included as the value of effectiveness cannot be calculated.

City	Random	Route use	Capacity	Katz	Betweenness	Flow	Eigenvector
Adelaide	0.119	0.641	0.512	0.146	0.212	0.190	0.146
Athens	0.244	0.486	0.683	0.088	0.209	0.134	0.090
Belfast	0.180	0.402	0.252	0.116	0.178	0.208	0.116
Bordeaux	0.177	0.583	0.570	0.089	0.197	0.143	0.090
Brisbane	0.120	0.284	0.414	0.205	0.199	0.116	0.206
Canberra	0.169	0.549	0.422	0.393	0.483	0.443	0.395
Detroit	0.201	0.410	0.366	0.181	0.186	0.246	0.189
Dublin	0.161	0.631	0.566	0.081	0.185	0.193	0.149
Helsinki	0.240	0.901	0.807	0.290	0.339	0.281	0.120
Kuopio	0.190	0.336	0.451	0.218	0.196	0.193	0.348
Lisbon	0.167	0.653	0.602	0.202	0.026	0.028	0.009
Melbourne	0.149	0.398	0.580	0.206	0.206	0.132	0.127
Nantes	0.198	0.554	0.626	0.217	0.142	0.084	0.075
Palermo	0.135	0.603	0.392	0.154	0.379	0.352	0.293
Prague	0.108	0.408	0.687	0.128	0.006	0.024	0.217
Rennes	0.125	0.525	0.477	0.123	0.223	0.116	0.188
Sydney	0.157	0.495	0.510	0.183	0.128	0.120	0.078
Toulouse	0.220	0.404	0.589	0.222	0.254	0.193	0.130
Venice	0.108	0.631	0.445	0.100	0.234	0.169	0.223
Winnipeg	0.238	1.008	0.727	0.107	0.603	0.461	0.407
$Mean \pm SE$	0.17 ± 0.01	0.54 ± 0.04	0.53 ± 0.03	0.17 ± 0.02	0.22 ± 0.03	0.19 ± 0.03	0.18 ± 0.02

4.4 Correlation of ranks in attack methods

In order to understand the magnitude of similarity of how our attack methods are ranking the routes, we analysed correlations between ranks of each route with different attack methods. Average Spearman rank-order correlation coefficient matrix of measures used for attack methods can be seen in Table 4. Katz centrality of static route-route networks does not seem to correlate well with any other method, which might be a result of said network having a very high density. Correlations between methods based on static methods, except Katz centrality, seem to correlate better with themselves than to route betweenness centrality (route use) and to nominal capacity. These two also seem to have a moderate correlation. The values for correlations among members of these two groups (nominal capacity and route use on one on one hand and centrality measures of static network minus Katz on the other; all in the 0.45 to 0.64 range) indicate that they are not redundant, not merely ordering routes the same way as another, but not completely irrelevant to each other at the same time. A previous survey of correlation of centrality measures on static network indicated similar results between multiple measures. The lower correlation coefficients between groups (with values between 0.13 and 0.34) indicate that there is fewer similarities between groups than within groups.

Table 4: Averages and standard deviations of Spearman rank-order correlation coefficient matrix of measures used for attack methods.

	Capacity	Betweenness	Flow	Katz	Eigenvector
Route Use	0.64 ± 0.15	0.34 ± 0.15	0.27 ± 0.13	-0.02 ± 0.09	0.22 ± 0.13
Capacity		0.25 ± 0.14	0.13 ± 0.17	-0.02 ± 0.06	0.15 ± 0.17
Betweenness			0.61 ± 0.14	0.01 ± 0.12	0.47 ± 0.16
Flow				0.01 ± 0.13	0.45 ± 0.20
Katz					-0.06 ± 0.11

4.5 Accessibility under attack and error

We can study the evolution of accessibility in more detail by plotting the distribution of number of accessible destinations starting from each source (as defined in 3.4) after performing an attack on the network. This way we do not simply see a single estimate for accessibility for each network; but the distributions of accessibilities show us how the accessibility varies within the city.

Figure 17 shows access for three different time-out values (5 hours, 2.5 hours and 1.25 hours) for two of the cities. Figure 18 shows changes to the distribution of access as a result of random failures and attacks for Helsinki. As more routes are removed we can see that more and more origins have a minimum level of accessibility while another group of nodes retain accessibility to most of the network. This effect is more pronounced with larger time-out values and with more routes removed. We

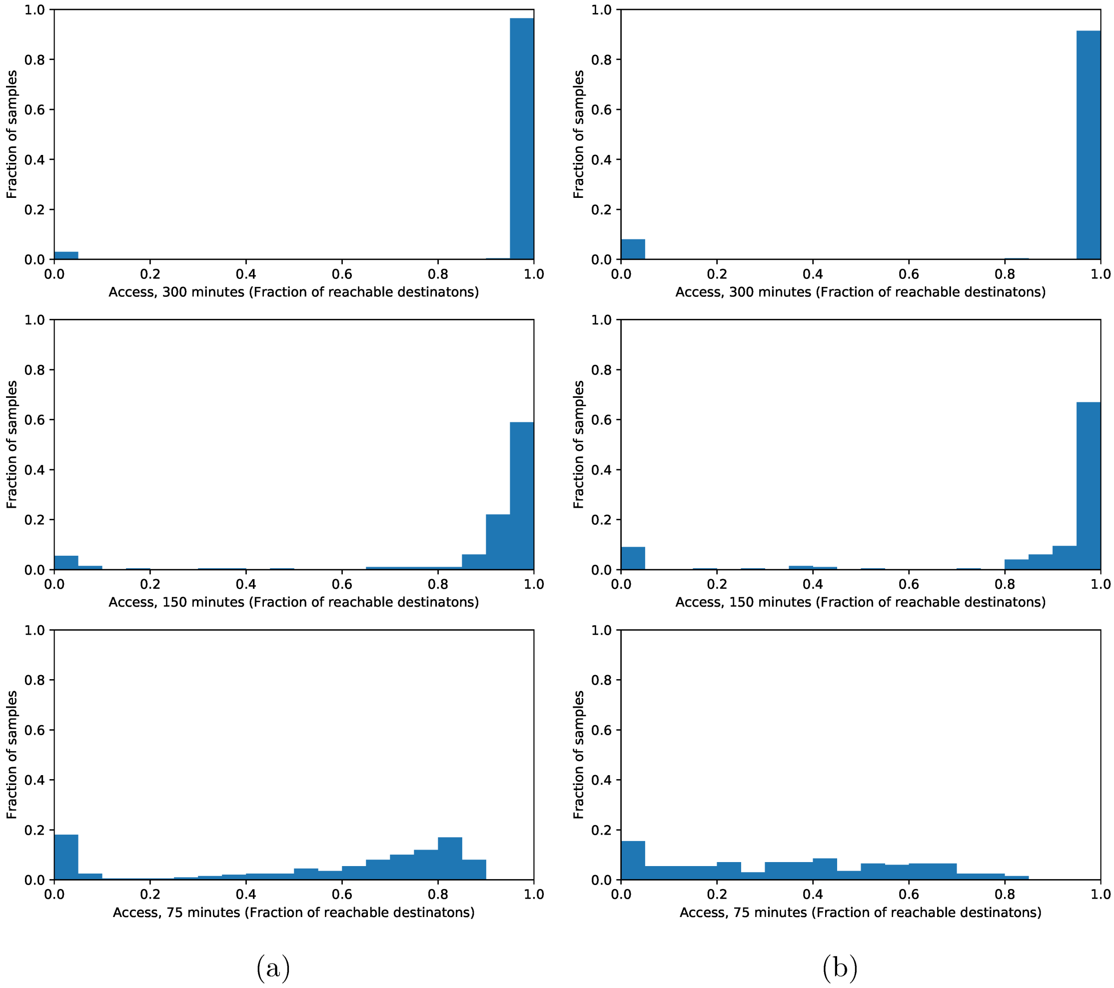
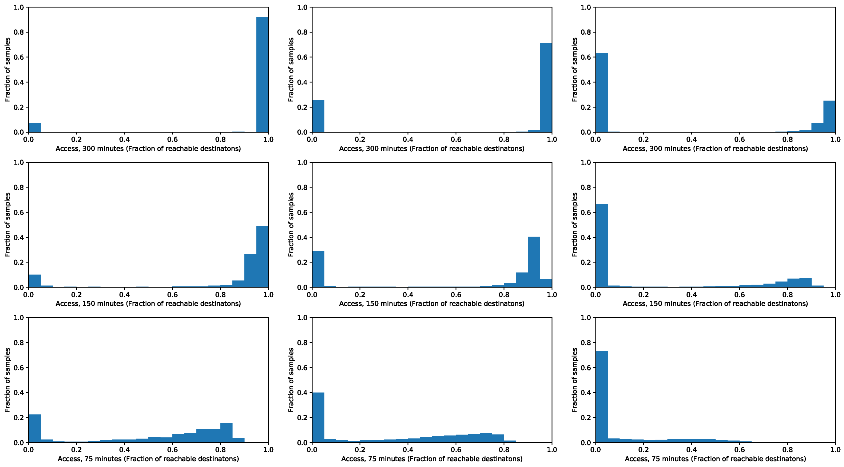
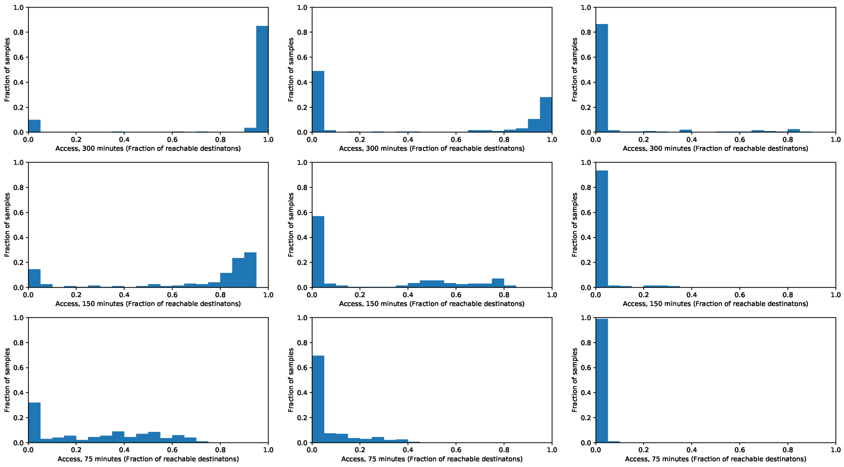


Figure 17: Distribution of access in Helsinki (8) and Adelaide (9) based on a sample of 200 origin stops without application of any attack or error.

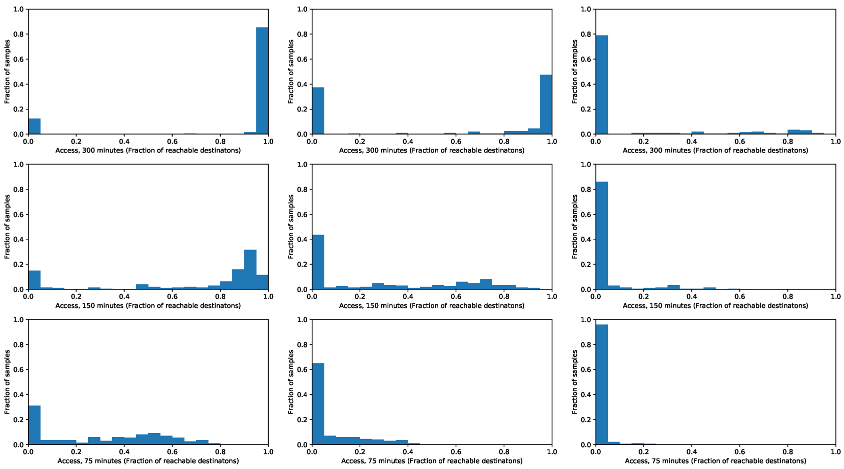
can also see that this situation of origins getting "disconnected" from the rest of the network is also more pronounced with capacity and route betweenness centrality attack methods. The bimodal nature of the distribution of accessibility makes it harder to measure accessibility of the whole network as a single number. There is a conceptual similarity between this situation and the evolution of component sizes in percolation theory, where the nodes that that some nodes have access to almost all the network are members of the temporal analogue of a "spanning cluster" and the nodes that have access only to a small fraction are members of smaller component. In a similar fashion, as the temporal spanning cluster breaks down the number of nodes in the smaller clusters increases.



(a) 12.4% Random (b) 48.8% Random (c) 86.8% Random



(d) 12.4% Route use (e) 48.8% Route use (f) 86.8% Route use



(g) 12.4% Capacity (h) 48.8% Capacity (i) 86.8% Capacity

Figure 18: Changes in distribution of access in Helsinki as a result of random failure (a, b and c), route betweenness centrality (d, e and f) and nominal capacity.

5 Discussion

In order to study the problem of connectivity in public transportation network, we modelled the public transportation system of each city as a temporal network where each stop is a node and each connection (with a vehicle or on-foot) between two stops is a link. Used methods of robustness, an extension of percolation theory on complex networks, to remove sets of links (in form of public transport routes) with different orders and study properties of the network as the number of removed routes increase.

As expected, some measures of ordering the routes were better at distinguishing more important routes, in terms of their effect on connectivity of the network. We found out that measures based on the temporal network performed much better than similar measures calculated on a static aggregated network designed for the study of connectivity of routes. In short, using various different centrality measures of an aggregate static network, the static route–route network, for devising attack methods did not produce attacks with effectiveness comparable to the temporal route betweenness method or the capacity based method or for that matter, significantly better than uniform random removal.

An interesting result of the various types of attack performed on the network is that for the case of two of our attack methods, nominal capacity and the route betweenness centrality calculated from temporal networks, even though different statistics used for determining attack order do not necessarily strongly correlate with each other (i.e. a correlation between ranks of a route in two attack methods is present but not very strong with a value of 0.64 ± 0.15) and even though order of these attacks are determined by very different methods, they more or less had the same effectiveness in increasing the travel times.

We also illustrated behaviour in the temporal network of public transportation similar to break-down of spanning cluster in percolation theory. As number of removed routes increased, more and more nodes seem to lose access to most of the other nodes of the network, only having access to a very small fraction of the nodes, while another group of nodes seem to retain access to most of the network. This shows the importance of developing and extending percolation theory to temporal networks. As connectivity in temporal network lacks certain properties of that in static, undirected networks, namely symmetricity, percolation in temporal networks would be more akin to that of directed network where connectivity of two nodes is not necessarily symmetric.

5.1 Temporal networks versus static networks

It has been shown that the speed of infection in a simple spreading process (SI model; where a susceptible node can become infected with a certain probability if it is in contact with another infected node) on temporal networks is affected by the temporal properties of the network [6, 7] e.g. the inhomogeneity in the frequency of the events between nodes as well as other types of temporal and structural correlations [8]. Given that our method of calculating shortest temporal paths is indistinguishable from SI model with origin node at original departure time being the only initially infected node and probability of infection set to 1, and that aggregation of the temporal network into a static network results in losing many of these correlations, it is plausible that doing so will result in different estimates for path lengths.

In previous studies of effects of temporal and structural correlations on spreading processes certain methods, e.g. reshuffling of timestamps or destinations of events, have been used to nullify such correlations and structures one by one and the effects studied [8]. It should be noted that we did not perform similar studies on this temporal network. In fact, some of the correlations identified for to be major be responsible for a part of this different spreading behaviour, such as burstiness of events [8, 66, 67], are not even likely to be present in the context of public transportation network due to mostly regular interval of departures and arrivals of vehicles at the stops.

But this possibility, that the network might behave differently with regard to spreading speed due to its temporal correlations and structures, combined with the observations that attack methods based on centrality measures from an aggregated network did not perform significantly better than random on impeding spreading process (as measured by travel times) point out to possible costs and dangers of using a static representation of a naturally temporal phenomenon as opposed to modelling it as a temporal network.

5.2 Alternative method of analysing robustness

In this study, we performed the measurement of the impact of an error or attack by selecting a sample of origins and calculating the average travel times (or average increase in the travel times) to all destinations. This method is very easy to grasp and straightforward but grows linearly more time-consuming as a function of number of samples.

Another method to achieve similar measurements, which we did not explore in this Thesis, is to form a weighted static event graph [68]. An event graph is a static, weighted, directed acyclic graph where each event is a node and two nodes are connected if they (a) share a node in the temporal network and (b) the event that leads to the common temporal node ends before the one that starts from it. The second condition is only required (and in fact only definable) for networks similar to transportation networks where events have an innate directionality. If both conditions apply, two events are connected in the event graph wherein direction of the link is from the event with earlier time-stamp to the event with the more recent time-stamp, and the weight of the link is determined by the time window between the end of the first events to the beginning of the second event. In a robustness scenario, after removing events that have to be removed as part of the attack or random failure, an upper bound on the number of possible reachable destination through events in each weakly connected components can be calculated based on the events in that component. This can be combined with a maximum waiting time threshold (δt; maximum allowed time for waiting for each transfer) as described in the original paper to further prune the event graph [68].

5.3 Real-world error tolerance and risk assessment

To have a better prospect of assessing damage and risk caused by failures in public transportation networks across different cities, it is important to derive failure probabilities and reduction of accessibility for all the different modes of failure in real-world which might not be studied in this report, e.g. effects and probability of vehicle failures. Given that a failure probability function can be calculated for components of transportation network of a city, one can redefine a more accurate "effectiveness of attack" quantity based on expected value of delays rather than the area under the delay curve as defined in this report in section 3.3.2.

It is important to take into account that this study focuses toward understanding the underlying structure of the network, as opposed to providing a tool to assess risks in case of an actual failure. There are many scenarios where a failure in a public transportation network might lead to results that cannot be assessed with the method provided here. As an example, our method does not take into account the maximum capacity of the transportation media; a total failure of the metro system in Helsinki might lead to an increase in the number of bus passengers that the system cannot easily handle. This can easily cause a cascade of failures scenario as the other media of transportation or alternative routes might fail to handle the excess load.

References

[1] Georg Zaklan, Frank Westerhoff, and Dietrich Stauffer. Analysing tax evasion dynamics via the ising model. *Journal of Economic Interaction and Coordination*, 4(1):1, 2009.

[2] W-X Zhou and Didier Sornette. Self-organizing ising model of financial markets. *The European Physical Journal B*, 55(2):175–181, 2007.

[3] Konstantin Klemm, Víctor M Eguíluz, Raúl Toral, and Maxi San Miguel. Nonequilibrium transitions in complex networks: A model of social interaction. *Physical Review E*, 67(2):026120, 2003.

[4] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440, 1998.

[5] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[6] Petter Holme and Jari Saramäki. Temporal networks. *Physics reports*, 519(3):97–125, 2012.

[7] Petter Holme. Modern temporal network theory: a colloquium. *The European Physical Journal B*, 88(9):234, 2015.

[8] Márton Karsai, Mikko Kivelä, Raj Kumar Pan, Kimmo Kaski, János Kertész, A-L Barabási, and Jari Saramäki. Small but slow world: How network topology and burstiness slow down spreading. *Physical Review E*, 83(2):025102, 2011.

[9] Vito Latora and Massimo Marchiori. Is the boston subway a small-world network? *Physica A: Statistical Mechanics and its Applications*, 314(1-4):109–113, 2002.

[10] Harold Soh, Sonja Lim, Tianyou Zhang, Xiuju Fu, Gary Kee Khoon Lee, Terence Gih Guang Hung, Pan Di, Silvester Prakasam, and Limsoon Wong. Weighted complex network analysis of travel routes on the singapore public transportation system. *Physica A: Statistical Mechanics and its Applications*, 389(24):5852–5863, 2010.

[11] W Li and X Cai. Empirical analysis of a scale-free railway network in china. *Physica A: Statistical Mechanics and its Applications*, 382(2):693–703, 2007.

[12] Parongama Sen, Subinay Dasgupta, Arnab Chatterjee, PA Sreeram, G Mukherjee, and SS Manna. Small-world properties of the indian railway network. *Physical Review E*, 67(3):036106, 2003.

[13] Ying Li, Wei Zhou, and Shi-jin Guo. An analysis of complexity of public transportation network in shanghai [j]. *Systems Engineering*, 1:006, 2007.

[14] YE Qing. Vulnerability analysis of rail transit based on complex network theory. *China Safety Science Journal*, 2(2):122–126, 2012.

[15] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308, 2006.

[16] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[17] Béla Bollobás. *Modern graph theory*, volume 184. Springer Science & Business Media, 2013.

[18] Mark Newman. *Networks: an introduction*. Oxford university press, 2010.

[19] Mark EJ Newman. Coauthorship networks and patterns of scientific collaboration. *Proceedings of the national academy of sciences*, 101(suppl 1):5200–5205, 2004.

[20] Wayne W Zachary. An information flow model for conflict and fission in small groups. *Journal of anthropological research*, 33(4):452–473, 1977.

[21] P Erdős and A Rényi. On random graphs i. *Publ. Math. Debrecen*, 6:290–297, 1959.

[22] Mark EJ Newman, Steven H Strogatz, and Duncan J Watts. Random graphs with arbitrary degree distributions and their applications. *Physical review E*, 64(2):026118, 2001.

[23] Trivik Verma, Nuno AM Araújo, and Hans J Herrmann. Revealing the structure of the world airline network. *arXiv preprint arXiv:1404.1368*, 2014.

[24] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power-law relationships of the internet topology. In *ACM SIGCOMM computer communication review*, volume 29, pages 251–262. ACM, 1999.

[25] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Internet: Diameter of the world-wide web. *nature*, 401(6749):130–131, 1999.

[26] Anna D Broido and Aaron Clauset. Scale-free networks are rare. *arXiv preprint arXiv:1801.03400*, 2018.

[27] Albert-László Barabási. Love is all you need; clauset's fruitless search for scale-free networks. Accessed at 22.04.2018 through URL https://www.barabasilab.com/post/love-is-all-you-need. Archived copy available at https://web.archive.org/web/20180422004242/https://uploads-ssl.webflow.com/58bcae2c9d6c401e73a26fed/5aa01d3e24eebb000199a0a2_loveisallyouneed.pdf.

[28] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[29] Dietrich Stauffer and Ammon Aharony. *Introduction to percolation theory*. CRC press, 1994.

[30] Reuven Cohen and Shlomo Havlin. Scale-free networks are ultrasmall. *Physical review letters*, 90(5):058701, 2003.

[31] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187. ACM, 2005.

[32] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[33] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

[34] Amnon Aharony and Dietrich Stauffer. *Introduction to percolation theory*. Taylor & Francis, 2003.

[35] Albert-László Barabási. *Network science*. Cambridge University Press, 2016.

[36] Jörg Stelling, Uwe Sauer, Zoltan Szallasi, Francis J Doyle, and John Doyle. Robustness of cellular functions. *Cell*, 118(6):675–685, 2004.

[37] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Error and attack tolerance of complex networks. *nature*, 406(6794):378–382, 2000.

[38] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P Gleeson, Yamir Moreno, and Mason A Porter. Multilayer networks. *Journal of complex networks*, 2(3):203–271, 2014.

[39] Lauri Kovanen, Kimmo Kaski, János Kertész, and Jari Saramäki. Temporal motifs reveal homophily, gender-specific patterns, and group talk in call sequences. *Proceedings of the National Academy of Sciences*, 110(45):18070–18075, 2013.

[40] Jean-Pierre Eckmann, Elisha Moses, and Danilo Sergi. Entropy of dialogues creates coherent structures in e-mail traffic. *Proceedings of the National Academy of Sciences of the United States of America*, 101(40):14333–14337, 2004.

[41] Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. Epidemic processes in complex networks. *Reviews of modern physics*, 87(3):925, 2015.

[42] Hartmut HK Lentz, Thomas Selhorst, and Igor M Sokolov. Unfolding accessibility provides a macroscopic approach to temporal networks. *Physical review letters*, 110(11):118701, 2013.

[43] Raj Kumar Pan and Jari Saramäki. Path lengths, correlations, and centrality in temporal networks. *Physical Review E*, 84(1):016105, 2011.

[44] Evangelia Pyrga, Frank Schulz, Dorothea Wagner, and Christos Zaroliagis. Efficient models for timetable information in public transportation systems. *Journal of Experimental Algorithmics (JEA)*, 12:2–4, 2008.

[45] André de Palma and Robin Lindsey. Optimal timetables for public transportation. *Transportation Research Part B: Methodological*, 35(8):789–813, 2001.

[46] Rainer Kujala, Christoffer Weckström, Miloš N Mladenović, and Jari Saramäki. Travel times and transfers in public transport: Comprehensive accessibility analysis based on pareto-optimal journeys. *Computers, Environment and Urban Systems*, 67:41–54, 2018.

[47] David Bohm. *The special theory of relativity*. WA Benjamin New York, 1965.

[48] Xi Lu, Hongping Wang, and Yong Deng. Evaluating the robustness of temporal networks considering spatiality of connections. *Chaos, Solitons & Fractals*, 78:176–184, 2015.

[49] Google Inc. General transit feed specification, 2017. Accessed 20.02.2018 through the URL https://developers.google.com/transit/gtfs/.

[50] SQLite Consortium. Sqlite.

[51] Christoffer Weckström, Rainer Kujala, Krzysztof Statkiewicz, Richard Darst, Milos Mladenovic, and Jari Saramäki. Deconet public transport network data repository, 2017. Accessed 20.02.2018 through the URL http://transportnetworks.cs.aalto.fi/browse/.

[52] Rainer Kujala, Christoffer Weckström, and Richard Darst. A collection of public transport network data sets for 25 cities, Feb 2018. The licensing terms are documented separately for each city.

[53] Google Inc. Extended gtfs route types, 2017. Accessed 20.02.2018 through the URL https://developers.google.com/transit/gtfs/reference/extended-route-types.

[54] Julian Dibbelt, Thomas Pajor, Ben Strasser, and Dorothea Wagner. Intriguingly simple and fast transit routing. In *International Symposium on Experimental Algorithms*, pages 43–54. Springer, 2013.

[55] Raymond C Browning, Emily A Baker, Jessica A Herron, and Rodger Kram. Effects of obesity and sex on the energetic cost and preferred speed of walking. *Journal of Applied Physiology*, 100(2):390–398, 2006.

[56] Belinda M Wu and Julian P Hine. A ptal approach to measuring changes in bus service accessibility. *Transport Policy*, 10(4):307–320, 2003.

[57] Julian Hine and Fiona Mitchell. Better for everyone? travel experiences and transport exclusion. *Urban studies*, 38(2):319–332, 2001.

[58] John Preston and Fiona Rajé. Accessibility, mobility and transport-related social exclusion. *Journal of Transport Geography*, 15(3):151–160, 2007.

[59] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007.

[60] Yaron Hollander and Ronghui Liu. Estimation of the distribution of travel times by repeated simulation. *Transportation Research Part C: Emerging Technologies*, 16(2):212–231, 2008.

[61] Nobuhiro Uno, Fumitaka Kurauchi, Hiroshi Tamura, and Yasunori Iida. Using bus probe data for analysis of travel time variability. *Journal of Intelligent Transportation Systems*, 13(1):2–15, 2009.

[62] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[63] Phillip Bonacich. Power and centrality: A family of measures. *American journal of sociology*, 92(5):1170–1182, 1987.

[64] Ulrik Brandes and Daniel Fleischer. Centrality measures based on current flow. In *Annual symposium on theoretical aspects of computer science*, pages 533–544. Springer, 2005.

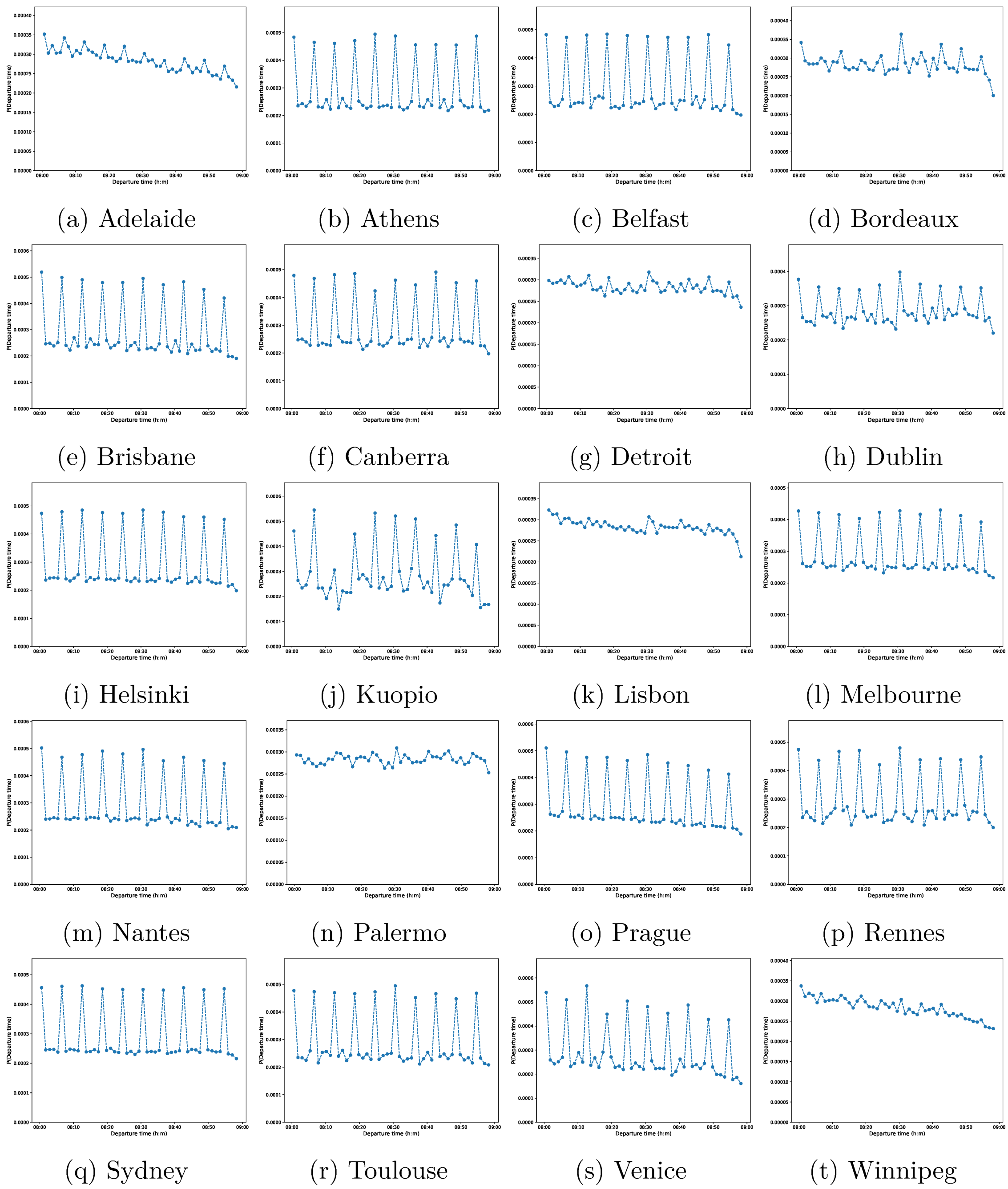
[65] Aleksi Manninen, Miska Peura, Eeva Rinta, and Anni Suomalainen. *Public Transport Planning Guidelines for HSL's Transport Services*, chapter 5, page 33. HSL Helsinki Region Transport, jun 2016. Accessed at 16.05.2017 through the URL https://www.hsl.fi/sites/default/files/uploads/joukkoliikenteen_suunnitteluohje_hsl-liikenteessa_2016.pdf.

[66] Dávid X Horváth and János Kertész. Spreading dynamics on networks: the role of burstiness, topology and non-stationarity. *New Journal of Physics*, 16(7):073037, 2014.

[67] Byungjoon Min, K-I Goh, and Alexei Vazquez. Spreading dynamics following bursty human activity patterns. *Physical Review E*, 83(3):036102, 2011.

[68] Mikko Kivelä, Jordan Cambe, Jari Saramäki, and Márton Karsai. Mapping temporal-network percolation to weighted, static event graphs. *arXiv preprint arXiv:1709.05647*, 2017.

A Figures for all cities



(a) Adelaide (b) Athens (c) Belfast (d) Bordeaux

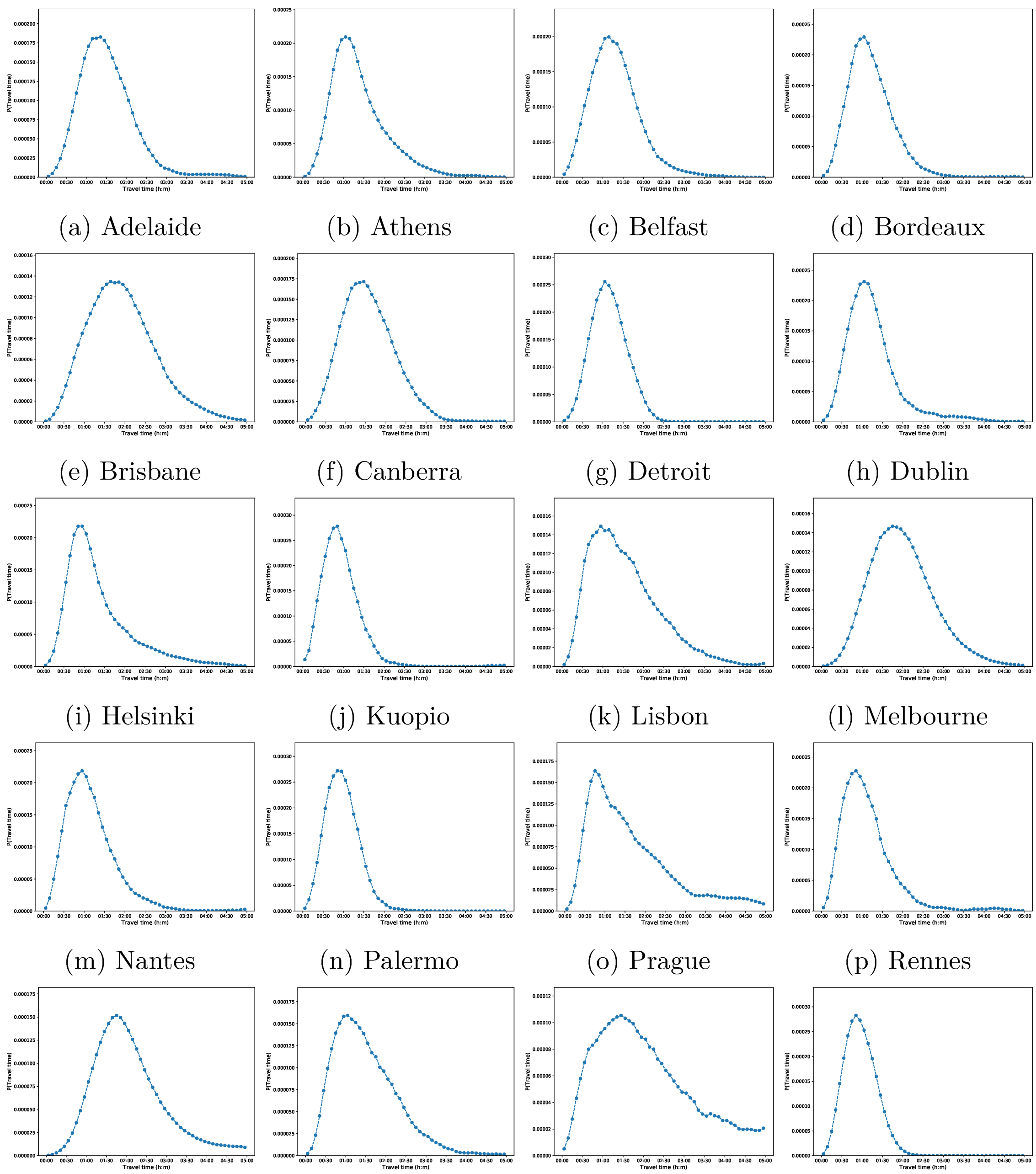
(e) Brisbane (f) Canberra (g) Detroit (h) Dublin

(i) Helsinki (j) Kuopio (k) Lisbon (l) Melbourne

(m) Nantes (n) Palermo (o) Prague (p) Rennes

(q) Sydney (r) Toulouse (s) Venice (t) Winnipeg

Figure A1: Probability density estimates for departure time of connections on all cities.



(a) Adelaide (b) Athens (c) Belfast (d) Bordeaux

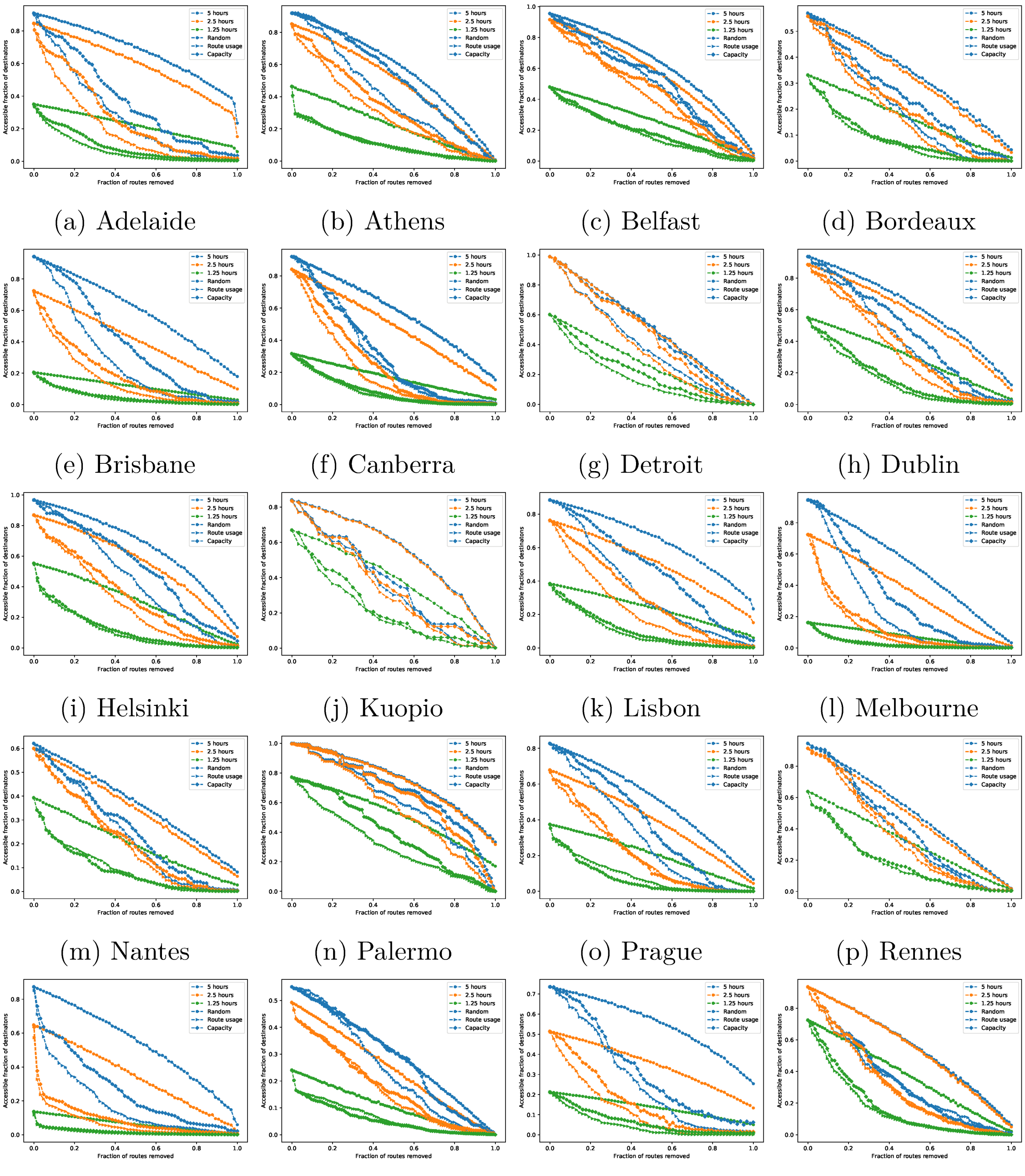
(e) Brisbane (f) Canberra (g) Detroit (h) Dublin

(i) Helsinki (j) Kuopio (k) Lisbon (l) Melbourne

(m) Nantes (n) Palermo (o) Prague (p) Rennes

(q) Sydney (r) Toulouse (s) Venice (t) Winnipeg

Figure A2: Probability density estimates average travel time without any attack or error for all cities.



(a) Adelaide (b) Athens (c) Belfast (d) Bordeaux

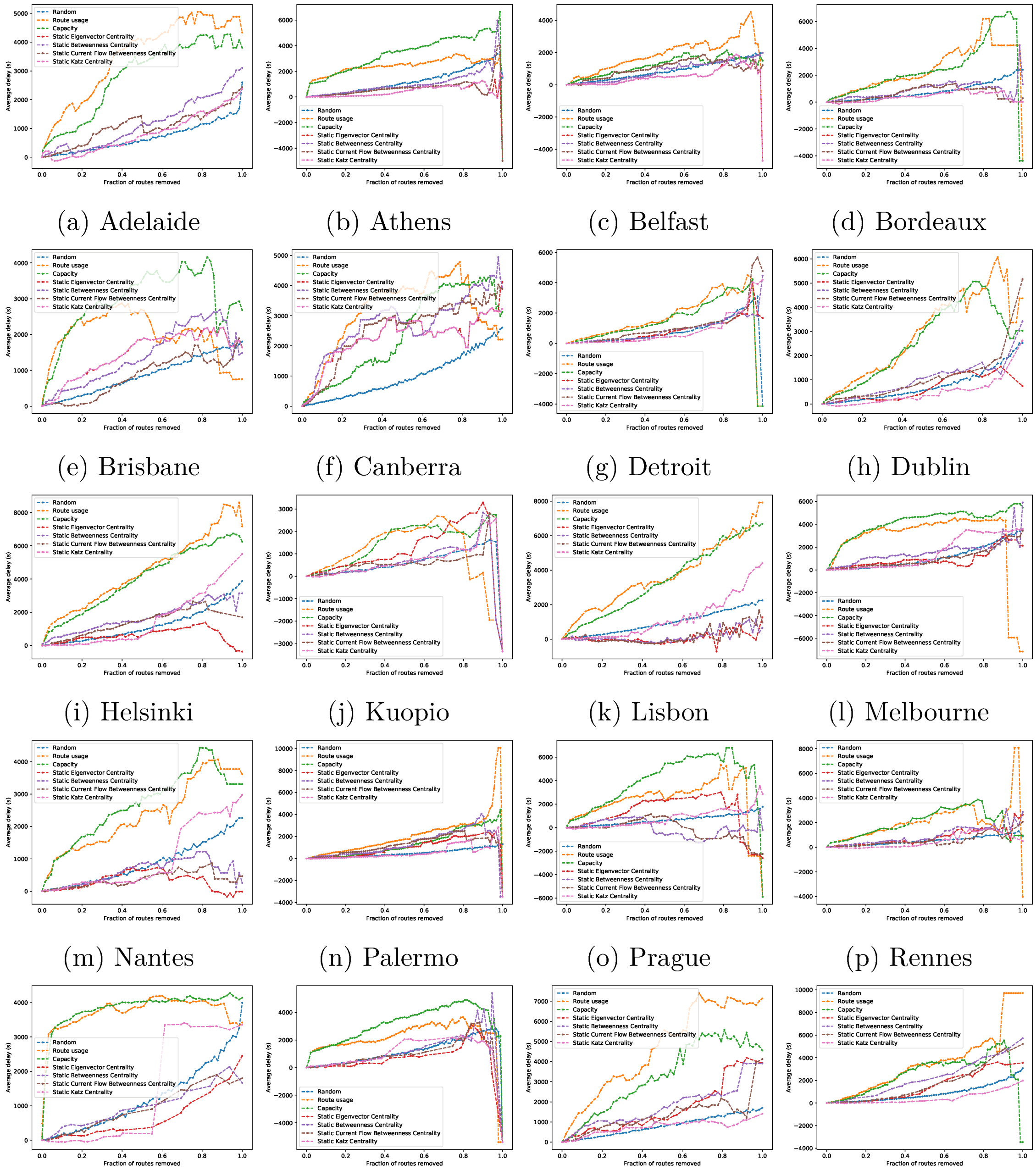
(e) Brisbane (f) Canberra (g) Detroit (h) Dublin

(i) Helsinki (j) Kuopio (k) Lisbon (l) Melbourne

(m) Nantes (n) Palermo (o) Prague (p) Rennes

(q) Sydney (r) Toulouse (s) Venice (t) Winnipeg

Figure A3: Decline in fraction of accessible possible destinations as a function of maximum acceptable time (colours) and method of attack (shapes).



(a) Adelaide (b) Athens (c) Belfast (d) Bordeaux

(e) Brisbane (f) Canberra (g) Detroit (h) Dublin

(i) Helsinki (j) Kuopio (k) Lisbon (l) Melbourne

(m) Nantes (n) Palermo (o) Prague (p) Rennes

(q) Sydney (r) Toulouse (s) Venice (t) Winnipeg

Figure A4: Changes of delay as a result of different attack methods. Travel times with very high fraction of routes removed are heavily influenced by walking distance to nearby stops as most other destinations become inaccessible due to five hours cut off as described in section 3.2.2